



# Attitude Determination of Highly Dynamic Fixed-wing UAVs

A MEMS-AHRS/GPS Integration

**Hugo José Dias Lopes**

Dissertação para a obtenção do Grau de Mestre em  
**Engenharia Aeroespacial**

## Júri

Presidente: Prof. João Manuel Lage de Miranda Lemos  
Orientador: Prof. Fernando Duarte Nunes  
Vogal: Prof. Paulo Jorge Coelho Ramalho Oliveira

**Outubro de 2011**



*À alegria da minha avó  
e à força de vontade do meu avô*



# Abstract

Taking advantage of the recent Micro-Electro-Mechanical Systems (MEMS), the global cost of the Unmanned Aerial Vehicles (UAVs) has been reduced. However, this reduction in size, power and price of the sensors comes at the expense of an increase in accuracy degradation making it more difficult to estimate the attitude of highly dynamic UAVs. Developing an efficient Attitude and Heading Reference System (AHRS) is then imperative where the integration of the Global Positioning System (GPS) and Inertial Navigation System (INS) can provide a more reliable and accurate AHRS.

In this thesis, the development of GPS/MEMS-INS systems specifically designed for attitude determination of fixed-wing UAVs is attempted and their performance evaluated. Two Extended Kalman Filters (EKF) are developed where the measurements equations are analytically solved in order to avoid the derivation of Jacobian matrices. This results in *Half* EKFs with simple measurements models. The algorithms make use of GPS-derived accelerations and airspeed sensors distinctly.

Regarding the simulation results, two flights were performed to test the two derived algorithms. Results show that the attitude of the UAV can be accurately estimated, with maximum error standard deviations rounding one degree. The EKF algorithm that attained best results in the simulations was tested with real flight data of a highly dynamic UAV and results show a consistent roll, pitch and yaw angles estimation. Comparisons were made with a commercial device (MTi-G Xsens) and the innovation sequences of the EKF algorithm support its reliability.

**Keywords:** UAV, MEMS, AHRS, GPS/INS integration, Kalman filtering.

# Resumo

Tirando partido dos recentes desenvolvimentos em Sistemas Micro-Eléctrico-Mecânicos (*Micro-Electro-Mechanical Systems* - MEMS), o custo global dos Veículos Aéreos Não Tripulados (*Unmanned Aerial Vehicles* - UAVs) tem vindo a ser reduzido. Contudo, esta redução em tamanho, consumo energético e preço dos sensores vem à custa de um aumento na degradação da precisão, tornando mais difícil estimar a atitude de UAVs altamente dinâmicos. Deste modo, desenvolver um Sistema de Referência de Atitude e Rumo (*Attitude and Heading Reference System* - AHRS) eficiente é imperativo, onde a integração do Sistema de Posicionamento Global (*Global Positioning System* - GPS) e do Sistema de Navegação Inercial (*Inertial Navigation System* - INS) conseguem proporcionar um AHRS mais fiável e preciso.

Nesta dissertação, o desenvolvimento de sistemas GPS/MEMS-INS, especificamente concebidos para a determinação da atitude de UAVs de asa-fixa, é efectuado e o seu desempenho avaliado. Dois Filtros de Kalman Generalizados (*Extended Kalman Filter* - EKF) são desenvolvidos onde as equações de observações são analiticamente resolvidas a fim de evitar a determinação das matrizes Jacobianais. Isto resulta em *Half EKFs* com modelos de observações mais simples. Um dos modelos faz uso das não tão comuns acelerações derivadas do GPS enquanto que o outro modelo usa um sensor de velocidade do ar (*airspeed sensor*) como sistema auxiliar.

Quanto aos resultados de simulação, foram efectuados dois voos para testar os algoritmos. Os resultados mostram que a atitude do UAV consegue ser determinada com precisão, com desvios padrão dos erros abaixo de um grau. O EKF que obteve melhores resultados nas simulações foi testado com dados de um voo real de um UAV altamente manobrável e os resultados mostram uma estimativa consistente dos ângulos de rolamento, picada e guinada. Foram efectuadas comparações com um aparelho comercial (*MTi-G Xsens*) e a sequência de inovações do EKF apoia a fiabilidade do algoritmo.

**Palavras-chave:** UAV, MEMS, AHRS, integração GPS/INS, Filtragem de Kalman.

# Acknowledgments

Without motivation, the achievement of our objectives is rather difficult or even impossible. Motivation is what TU Delft is full of. The inspiring work environment that I experienced and excellent people I met made me attain my objectives in order to complete my Master Double Degree TU Delft - IST Lisbon.

I would like to express my gratitude to my Portuguese supervisor Prof. Fernando Duarte Nunes for his advice, assistance and constant presence during all my developments. Additionally, I would like to thank Dr. ir. Q. P. Chu for his continuous support and amazing capacity to encourage me along the thesis work. I want to express my sincere appreciation to Dr. ir. E. van Kampen for his help and important remarks that improved the quality of this thesis. Finally, I would like to thank Prof. Dr. ir. J. A. Mulder for the pleasant and inspiring environment that I worked during all the these months.

A special thank goes to the *Simona rooms* where I met fantastic people that made my everyday work more enjoyable. Obtaining proper simulated flight data would have been impossible without Gwan Pieters, and I just have to thank his patience!

This year abroad at The Netherlands would not have been amazing without all the international people I met, specially at *Marcushof*, and made me feel really fortunate. A special thank has to go to all the other Portuguese students that present energy, intelligence and motivation everyday with the aim to accomplish their objectives with the highest quality possible. These strong factors around me helped to make this thesis attainable.

Finally, I would like to express my gratitude to all the remaining people that are very close to me. During hard times we may think that there are mountains that we can not climb. However, with their help I could go beyond my limits.

Lisboa, Instituto Superior Técnico  
September 26, 2011

Hugo J. Dias Lopes

# Contents

<b>Acronyms</b>	<b>xiv</b>
<b>List of Symbols</b>	<b>xv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Thesis Background . . . . .	1
1.1.1 Attitude and Heading Reference Systems . . . . .	1
1.2 Research Objectives . . . . .	2
1.3 Thesis Outline . . . . .	2
<b>2 Fundamentals</b>	<b>5</b>
2.1 Reference Frames . . . . .	5
2.1.1 Earth-centered, Earth-fixed reference frame . . . . .	5
2.1.2 Body-fixed reference frame . . . . .	5
2.1.3 Navigational reference frame . . . . .	6
2.1.4 Aerodynamic reference frame . . . . .	7
2.1.5 Transformations between Reference Frames . . . . .	7
2.2 Global Position System . . . . .	11
2.2.1 Principle of the Global Positioning System . . . . .	11
2.2.2 GPS Velocity . . . . .	12
2.2.3 GPS Acceleration . . . . .	13
2.2.4 GPS Sources of Error . . . . .	13
2.3 Inertial Navigation . . . . .	14
2.3.1 Low Cost MEMS-based IMUs . . . . .	15
2.3.2 Inertial Sensor Errors . . . . .	18
2.4 Additional Aircraft Sensors . . . . .	19
2.4.1 Altimetry Sensors . . . . .	19
2.4.2 Airspeed Sensor . . . . .	19
2.4.3 Angle of Attack and Angle of Sideslip Sensors . . . . .	20
2.4.4 Magnetometers . . . . .	21
<b>3 GPS/INS Integration</b>	<b>23</b>
3.1 GPS/INS Integration Architectures . . . . .	23
3.1.1 Loosely Coupled Integration . . . . .	24
3.1.2 Tightly Coupled Integration . . . . .	25
3.1.3 Other Integration Methods . . . . .	25

3.1.4	Comparison Between the Architectures . . . . .	26
3.2	Kalman Filtering . . . . .	27
3.2.1	Introduction . . . . .	27
3.2.2	The Discrete Kalman Filter . . . . .	27
3.2.3	The Kalman Filter Formulation . . . . .	29
3.2.4	Kalman Filter Tuning Parameters . . . . .	30
3.2.5	Kalman Filter Discretization Issues . . . . .	30
3.2.6	Extended Kalman Filter . . . . .	32
3.2.7	Reliability Testing . . . . .	33
<b>4</b>	<b>Modeling and Implementation</b>	<b>35</b>
4.1	Sensors Modeling . . . . .	35
4.1.1	Inertial Measuring Units (IMU) Modeling . . . . .	35
4.1.2	GPS Modeling . . . . .	38
4.2	Filter Implementation . . . . .	39
4.2.1	Equations of Motion . . . . .	40
4.2.2	Process Model . . . . .	41
4.2.3	Measurement Model 1 . . . . .	43
4.2.4	Measurement Model 2 . . . . .	46
4.2.5	Other implementation issues . . . . .	49
<b>5</b>	<b>Tests and Results</b>	<b>51</b>
5.1	Simulation Environment . . . . .	51
5.1.1	Aeronautical Simulation Library . . . . .	51
5.1.2	GPS/INS Synchronization . . . . .	52
5.1.3	Sensors Characteristics . . . . .	52
5.1.4	Simulation Flights . . . . .	53
5.1.5	Other Simulation Issues . . . . .	56
5.2	Simulation Results . . . . .	57
5.2.1	Results of EKF <sub>1</sub> . . . . .	57
5.2.2	Results of EKF <sub>2</sub> . . . . .	63
5.2.3	GPS outages . . . . .	70
5.2.4	Simulation Summary . . . . .	72
5.3	Real Flight . . . . .	72
5.3.1	Real Flight Issues . . . . .	73
5.3.2	Real Flight Results . . . . .	74
<b>6</b>	<b>Conclusions and Recommendations</b>	<b>77</b>
6.1	Conclusions . . . . .	77
6.2	Recommendations . . . . .	78
	<b>Bibliography</b>	<b>80</b>
<b>A</b>	<b>Stochastic Processes</b>	<b>84</b>
A.1	White Noise . . . . .	84
A.2	Random Walk . . . . .	84
A.3	Random Constant . . . . .	84
A.4	Gauss-Markov Process . . . . .	85
<b>B</b>	<b>Gaussian Distribution Verification</b>	<b>86</b>

<b>C</b>	<b>Simulation Flights Data</b>	<b>88</b>
<b>D</b>	<b>Additional Simulation Results</b>	<b>92</b>
D.1	EKF <sub>1</sub> and EKF <sub>2</sub> . . . . .	92
D.2	GPS Outages . . . . .	96
<b>E</b>	<b>Real Data</b>	<b>97</b>
E.1	MTi-G - Xsens Device . . . . .	97
E.2	Additional Data of Real Flight . . . . .	98
E.3	Tunning Parameters . . . . .	99

# List of Figures

2.1	Earth-centered, Earth-fixed reference frame $\mathcal{F}_E$ . . . . .	6
2.2	Body-fixed reference frame $\mathcal{F}_B$ . . . . .	6
2.3	Navigational reference frame $\mathcal{F}_N$ . . . . .	6
2.4	Aerodynamic reference frame $\mathcal{F}_A$ . . . . .	7
2.5	Example of a transformation 3-1-3, from $\mathcal{F}_{C_1}$ to $\mathcal{F}_{C_2}$ . . . . .	8
2.6	Definition of course angle $\chi$ . . . . .	10
2.7	Definition of the angle of attack $\alpha$ , climb angle $\gamma$ and pitch angle $\theta$ . . . . .	10
2.8	GPS segments: Space, Control and User (source: <a href="http://connet.us/gps.htm">http://connet.us/gps.htm</a> ). . . . .	11
2.9	Concepts of INS and IMU (adapted from (Schultz, 2006)). . . . .	14
2.10	Inertial Gimbal and Strapdown systems, from (Grewal, Weill, & Andrews, 2001). . . . .	15
2.11	MEMS technology: accelerometer and gyroscope, from (Barbour, 2004). . . . .	15
2.12	Conventional mechanical accelerometer, from (Titterton & Weston, 2004). . . . .	16
2.13	Types of MEMS accelerometers from (Barbour, 2004). . . . .	16
2.14	Difference between gravitational acceleration and vehicle acceleration, adapted from (Schultz, 2006). . . . .	17
2.15	Conventional mechanical gyroscope (source: <a href="http://www.gyroscopes.org">www.gyroscopes.org</a> ). . . . .	18
2.16	Fiber Optic Gyroscope (source: <a href="http://www.alcielo.com">www.alcielo.com</a> ). . . . .	18
2.17	Two types of MEMS gyroscopes from (Barbour, 2004). . . . .	18
2.18	Aneroid static air pressure meter: a vacuum membrane changes its shape as a function of static pressure (M. Mulder, 2010). . . . .	20
2.19	Pitot-static probe: changes in membrane shape are related to differences between total and static pressure (M. Mulder, 2010). . . . .	20
2.20	Angle of attack sensor (Pallett, 1996). . . . .	20
2.21	The Magnetic Dip, from (M. Mulder, 2010). In The Netherlands it is about $66.7^\circ$ . . . . .	21
3.1	Loosely coupled GPS/INS integration concept, from (Schmidt, 2004). . . . .	24
3.2	Tightly coupled GPS/INS integration concept, from (Schmidt, 2004). . . . .	25
5.1	IMU and GPS output frequencies are different and require synchronization. . . . .	52
5.2	Example of a gyroscope signal after noise and bias introduction from Table 5.1. . . . .	53
5.3	3D position of Flight 1. Initial position: [0, 0, 140]m. . . . .	54
5.4	3D position of Flight 2. Initial position: [0, 0, 40]m. . . . .	55
5.5	Estimation of $\phi$ , $\theta$ and $\psi$ , and differences between real and estimated angles, with EKF <sub>1</sub> , Flight 1. . . . .	58
5.6	Estimation of $b_p$ , $b_q$ and $b_r$ , and differences between real and estimated bias with EKF <sub>1</sub> , Flight 1. . . . .	59

5.7	Innovation of the three Euler angles with EKF <sub>1</sub> for Flight 1. The red line represents the standard deviation of the Innovation. . . . .	60
5.8	Estimation of $\phi$ , $\theta$ and $\psi$ , and differences between real and estimated angles, with EKF <sub>1</sub> , Flight 2. . . . .	61
5.9	Estimation of $b_p$ , $b_q$ and $b_r$ , and differences between real and estimated bias with EKF <sub>1</sub> , Flight 2. . . . .	62
5.10	Differentiation of the noisy airspeed measurements by two methods. . . . .	64
5.11	Estimation of $\phi$ , $\theta$ and $\psi$ , and differences between real and estimated angles, with EKF <sub>2</sub> , Flight 1. . . . .	65
5.12	Estimation of $b_p$ , $b_q$ and $b_r$ , and differences between real and estimated bias with EKF <sub>2</sub> , Flight 1. . . . .	66
5.13	Innovation of the three Euler angles with EKF <sub>2</sub> for Flight 1. The red line represents the standard deviation of the Innovation. . . . .	67
5.14	Estimation of $\phi$ , $\theta$ and $\psi$ , and differences between real and estimated angles, with EKF <sub>2</sub> , Flight 2. . . . .	68
5.15	Estimation of $b_p$ , $b_q$ and $b_r$ , and differences between real and estimated bias with EKF <sub>2</sub> , Flight 2. . . . .	69
5.16	States estimation upon GPS outages during $t \in [60, 80]$ s (Flight 1). . . . .	71
5.17	Easystar UAV with 1.4m wingspan and 0.68kg of (empty) weight. . . . .	73
5.18	Estimation of the attitude angles $\phi$ , $\theta$ and $\psi$ for the real flight. . . . .	74
5.19	Innovation of the three Euler angles for the real flight. The red line represents the standard deviation of the Innovation. . . . .	76
B.1	Verification of a Gaussian fitting for the noise of the measurements vector. . . . .	87
C.1	Data from Flight 1: accelerometers, GPS velocity and gyroscopes measurements. . . . .	88
C.2	Data from Flight 1: angle of sideslip, angle of attack and airspeed. . . . .	89
C.3	Data from Flight 1: altitude. . . . .	89
C.4	Data from Flight 2: accelerometers, GPS velocity and gyroscopes measurements. . . . .	90
C.5	Data from Flight 2: angle of sideslip, angle of attack and airspeed. . . . .	91
C.6	Data from Flight 2: altitude. . . . .	91
D.1	Pitch angle estimation with EKF <sub>1</sub> for Flight 2, with verification of Eq. (2.13). . . . .	92
D.2	Estimation of $\phi$ , $\theta$ and $\psi$ , and differences between real and estimated angles, with EKF <sub>1</sub> for Flight 2 with inclusion of Eq. (5.9). . . . .	93
D.3	Estimation of $b_p$ , $b_q$ and $b_r$ , and differences between real and estimated bias with EKF <sub>1</sub> for Flight 2 with inclusion of Eq. (5.9). . . . .	94
D.4	Innovation of the three Euler angles with EKF <sub>1</sub> for Flight 2. The red line represents the standard deviation of the Innovation. . . . .	95
D.5	Innovation of the three Euler angles with EKF <sub>2</sub> for Flight 2. The red line represents the standard deviation of the Innovation. . . . .	95
D.6	Attitude angles estimation with GPS outages during $t \in [60, 80]$ (Flight 1). . . . .	96
D.7	Estimation of the bias of the gyroscopes with GPS outages during $t \in [60, 80]$ (Flight 1). . . . .	96
E.1	Real Flight Data (LOG00060) - North-East position (GPS) and Altitude (GPS-Barometer). . . . .	98
E.2	Real Flight Data (LOG00060) - accelerometers and gyroscopes measurements. . . . .	98
E.3	Real Flight Data (LOG00060) - GPS velocity NED. . . . .	99
E.4	Estimation of the gyroscopes' bias for the Real Flight. . . . .	99

# List of Tables

3.1 Comparison between Inertial and Satellite Navigation Systems. . . . .	23
5.1 Noise and bias introduced in the simulation sensors. . . . .	53
5.2 Numerical results of EKF <sub>1</sub> for Flight 1. . . . .	60
5.3 Innovation results of EKF <sub>1</sub> for Flight 1. . . . .	60
5.4 Numerical results of EKF <sub>1</sub> for Flight 2 with and without inclusion of Eq. (5.9) in the algorithm. . . . .	63
5.5 Innovation results of EKF <sub>1</sub> for Flight 2. . . . .	63
5.6 Numerical results of EKF <sub>2</sub> for Flight 1. . . . .	67
5.7 Innovation results of EKF <sub>2</sub> for Flight 1. . . . .	67
5.8 Numerical results of EKF <sub>2</sub> for Flight 2. . . . .	70
5.9 Innovation results of EKF <sub>2</sub> for Flight 2. . . . .	70
5.10 Deviation of the estimated states from the real states at $t = 79.99\text{s}$ . . . . .	70
5.11 Numerical results of Real Flight comparing with the MTi-G device. . . . .	75
5.12 Innovation results of Real Flight. . . . .	75
E.1 Calibrated inertial data performance for a MTi-G device with standard configuration. . .	97

# Acronyms

**AHRS** Attitude and Heading Reference System

**ECEF** Earth-centered, Earth-fixed

**EKF** Extended Kalman Filter

**EOM** Equations of Motion

**GLONASS** GLObal NAVigation Satellite System

**GNSS** Global Navigation Satellite System

**GPS** Global Positioning System

**IMU** Inertial Measurement Units

**INS** Inertial Navigation System

**ISA** International Standard Atmosphere

**KF** Kalman Filter

**MAV** Micro Aerial Vehicles

**MEMS** Micro-Electro-Mechanical Systems

**NED** North-East-Down

**PPS** Pulse-Per-Second

**TAS** True Airspeed

**UAV** Unmanned Aerial Vehicles

# List of Symbols

## Greek Symbols

$\alpha$	Angle of attack
$\beta$	Sideslip angle
$\chi$	Course angle
$\delta(\cdot)$	Dirac delta function
$\delta_{ij}$	Kronecker delta function
$\Delta t$	Sampling time / Time interval
$\varepsilon$	Measurement noise
$\varepsilon_c$	Correlated noise
$\varepsilon_d$	Dither noise
$\varepsilon_q$	Quantization noise
$\varepsilon_r$	Random walk noise
$\varepsilon_w$	White noise
$\gamma$	Flight path angle
$\lambda$	Scalar weight for the gyroscopes' bias estimator
$\nu$	Innovation
$\underline{\omega}$	3-axis true gyroscopes values vector
$\Omega_{Earth}$	Rotational rate of the Earth
$\Phi$	System Transition matrix
$\phi$	Roll angle
$\varphi$	General angle
$\Psi$	Input distribution matrix
$\psi$	Yaw angle
$\sigma$	Standard deviation
$\sigma_\nu$	Covariance of Innovation
$\tau_c$	Gauss-Markov correlation time
$\theta$	Pitch angle

## Roman Symbols

$a$	Acceleration
$\underline{A}$	3-axis true accelerations values vector
$a_{GPS}$	GPS-derived acceleration
$B$	Linear Input matrix
$b_0$	Initial sensor bias
$b$	Sensor bias
$b_R$	Sensor bias-drift

$D$	Feedforward matrix
$\underline{e}$	Error vector between the true and the estimated states
$E$	Expected value operator
$\mathcal{F}$	Referential Frame
$F_x$	Jacobian of the Dynamics equation
$f$	General frequency
$F$	(Linear) System/Process matrix
$F$	General force
$\underline{F}$	External forces (vector) applied to the body
$f_s$	Scale factor error function
$\underline{f}$	Specific force
$\underline{f}$	System (nonlinear) dynamics
$g$	Gravitational acceleration
$G$	System noise input matrix
$H$	(Linear) Observation matrix
$H_x$	Jacobian of the Observer equation
$I$	Identity matrix
$K$	Kalman Gain
$m$	Mass
$P_0$	Initial Error Covariance matrix
$P$	Error covariance matrix
$p$	$X_B$ -axis gyroscope (rotational rate) value
$Q$	Process noise covariance matrix
$Q_d$	Discretized Process noise covariance matrix
$q$	$Y_B$ -axis gyroscope (rotational rate) value
$R$	General Rotation Matrix
$R_{A/B}$	Rotation Matrix from Aerodynamic to Body-fixed reference frame
$R_{B/N}$	Rotation Matrix from Navigational to Body-fixed reference frame
$r$	$Z_B$ -axis gyroscope (rotational rate) value
$R$	Measurements noise covariance matrix
$R_p$	Package misalignment matrix
$R_s$	Sensor-to-sensor misalignment matrix
$\underline{u}$	Input vector
$u$	$X$ -axis velocity
$V_a$	True airspeed
$\underline{v}$	Measurements noise vector
$v$	$Y$ -axis velocity
$w$	White noise
$\underline{w}$	System/Process noise vector
$w$	$Z$ -axis velocity
$X$	X-axis of a referential frame
$\underline{x}$	States vector
$x_0$	Initial condition of the states vector
$Y$	Y-axis of a referential frame
$Z$	Z-axis of a referential frame
$\underline{z}$	Measurements vector

## Subscripts

$A$  Quantity in the Aerodynamic reference frame

$A$	Quantity relative to an accelerometer
$b_p$	Quantity relative to the bias of the $p$ gyroscope
$b_q$	Quantity relative to the bias of the $q$ gyroscope
$b_r$	Quantity relative to the bias of the $r$ gyroscope
$B$	Quantity in a Body-fixed reference frame
$d$	Discretized quantity
$E$	Quantity in an Earth-Centered, Earth-Fixed reference frame
$GPS$	Quantity relative to GPS
$m$	Measured quantity
$\omega$	Quantity relative to a gyroscope
$p$	Quantity relative to the $p$ gyroscope
$q$	Quantity relative to the $q$ gyroscope
$r$	Quantity relative to the $r$ gyroscope
$x$	Quantity along X-axis
$y$	Quantity along Y-axis
$z$	Quantity along Z-axis

## Others

- $\underline{\square}$  Vector notation
- $\dot{\square}$  First time derivative
- $\hat{\square}$  Estimated value
- $\square^T$  Transpose of a vector or matrix
- $\square^*$  Nominal value



# Chapter 1

## Introduction

In this first chapter, the thesis subject is going to be presented. The thesis background and related work is presented in Section 1.1. Then, the problem statement and purposed objectives are discussed in Section 1.2. Finally, the structure of the thesis report can be consulted in Section 1.3.

### 1.1 Thesis Background

The Global Positioning System (GPS) and Inertial Navigation System (INS) data fusion is not a new subject and many research have been carried, (Caron, Duflos, Pomorski, & Vanheeghe, 2006; Gebre-Egziabher, Hayward, & Powell, 2004; Petovello, 2003). This GPS/INS integration have been implemented in the last years in several types of applications such as ships, aircraft, and submarine navigation systems. The INS can provide continuous and reliable navigation determination. However, the main drawback of this system is that its error increases over time. On the other hand, the GPS can be used to correct these errors, as an aiding system because it provides long-term stability with high accuracy (Kim, Sukkarieh, & Wishart, 2006). Thus, this fusion between the GPS and INS strive to achieve the following (Schultz, 2006):

- Acceptable accuracy level with the possibility to keep decent accuracy over the time.
- Continuous and reliable navigation determination (e.g., position, velocity and attitude).

Recent interest in design and flight of small Unmanned Aerial Vehicles (UAV) and Micro Aerial Vehicles (MAV) has prompted research into control and navigation of this type of vehicles. The potential uses of UAVs in the civil industry are extensive, such as (Sarris, 2001): border interdiction, search and rescue, wild fire suppression, industrial applications (e.g., agriculture, surveillance), etc. However, in order to become successful, the global cost of the UAVs has to be affordable to the civilian market. This has led to large strides in sensor cost reduction where the advent of the low-cost Micro-Electro-Mechanical Systems (MEMS) accelerometers and gyroscopes took place in the last years, (Kim et al., 2006; Barbour, 2004).

#### 1.1.1 Attitude and Heading Reference Systems

When the final product of the GPS/INS integration algorithm is the attitude solution it is referred to as Attitude and Heading Reference System (AHRS), (Gleason & Gebre-Egziabher, 2009). The AHRS can be seen as an integrated sensor system providing roll, pitch and heading angles with the use accelerometers, rate gyros, and magnetometers (Chu, 2011). The obtained attitude and heading angles are then used for aircraft flight control and display. With the integration of the GPS as an aiding system, the final outcome is an integrated AHRS/GPS system.

The basis of an UAV/MAV autonomous flight and mission execution is attitude control, while the premise of the attitude control is the real-time access to the reliable, dynamic and accurate attitude information of an aircraft (Wang, Li, & He, 2011). Restricted by the weight, size and power, the UAVs/MAVs can not carry traditional attitude measurement systems. Consequently, several AHRS structures with GPS/MEMS-INS integration have been investigated, especially making use of a Kalman filter (Shin, 2005; Caron et al., 2006; Cheng, Zhaoying, & Xu, 2008).

An inexpensive gyro-free attitude determination system that makes use of accelerometers, magnetometers and GPS, as primary sensors, have been studied in (Gebre-Egziabher, Klkaim, Powell, & Parkinson, 2000). Additionally, an algorithm that takes advantage of the relatively accurate GPS  $L_1$  carrier phase measurements to derive the acceleration of the user was developed in (Y. Li, Dempster, Li, Wang, & Rizos, 2006). Moreover, GPS-based attitude determination obtained by differencing signals from multiple antennas has also been a research topic, investigated by many researchers, such as (Hayward, Gebre-Egziabher, Schwall, Powell, & Wilson, 1997; Gebre-Egziabher, Hayward, & Powell, 1998). An approach that did not make use of inertial sensors was studied in (Kornfeld, Hansman, & Deyst, 1998). Instead, a kinematic model of an aircraft along with the GPS position and velocity measurements derived from a single GPS antenna was used to generate a so called *pseudo-attitude*. More recently, *vector matching* algorithms that makes use of the Earth's magnetic and gravity field vectors as observations, and GPS as an aiding system, have been derived and successfully implemented (Gebre-Egziabher & Elkaim, 2008).

## 1.2 Research Objectives

The main objective of this thesis is to investigate the performance of an integrated low-cost GPS/MEMS-INS system specifically designed for (fixed-wing) UAVs airborne applications. Since the focus will be in the attitude determination it could also be referred as GPS/MEMS-AHRS system.

The use of air data sensors, such as airspeed sensors, but more importantly the usage of GPS-derived accelerations in this integration systems are open research fields and will be discussed. Additionally, upon the Kalman Filter implementation, the derivation of the Jacobian matrices for both process and measurement equations is nontrivial in most applications, and quite often leads to significant implementation difficulties. A novel approach in order to overcome this problem will be attempted, by analytically solving the measurements equations.

Simple, efficient, low-cost and low-power demanding are the characteristics desired for the final GPS/INS integration derived algorithms. The main objectives steps of this thesis are then:

- Research about the more adequate sensors to use in the small UAVs in order to develop a GPS/INS integration system (possibly with air data sensors also).
- Derivation of simple and fast (Extended) Kalman filter algorithms, where analytical solutions are attempted.
- Simulate the derived algorithms and test their limits. Provide comparison between them.
- Application of the developed algorithms to an UAV's real flight data under high maneuvering scenarios.

## 1.3 Thesis Outline

This thesis consists of six chapters and five appendices which are organized as follows.

The *Chapter 2* describes the theory background necessary to understand the basics of this thesis. The various reference frames required and a brief description about the transformation between them are

provided. The Global Positioning System principles, the GPS Velocity and Acceleration, and the main sources of error of this type of navigation system are presented as well. Following on the descriptions, the Inertial Navigation System is also explored where its main characteristics are discussed. The accelerometers and gyroscopes principles and state-of-art are described. Finally, additional aircraft sensors, such as airspeed sensors, are also presented.

In *Chapter 3* the different types of GPS/INS integration structures, their advantages and drawbacks are presented. Comparison between them leads to the chosen type of GPS/INS integration architecture. After that, the Kalman Filtering is explored and its generalization to nonlinear models, discretization, tuning and reliability issues are discussed.

Then, the *Chapter 4* considers the necessary modeling and implementation to finally arrive to a GPS/INS estimation algorithm. The modeling of the sensors is performed where the Inertial Measuring Units and GPS noise and biases are discussed and models are presented. The most important derivations are performed in this chapter where the Equations of Motion, the Process and Measurement models of the Kalman filter, and other implementation issues are addressed. In the end of the chapter, two Extended Kalman Filters are derived.

The *Chapter 5* presents the simulation and real flights results, and conclusions are drawn about the derived GPS/INS integration algorithms. The chapter begins with the description of the simulation environment where the sensor characteristics, the GPS/INS synchronization issue and the simulation profiles are discussed. Then, the algorithms are tested and compared with two simulation flights. Finally, one of the algorithms is tested with real flight data of an UAV.

The conclusions and recommendations for future work are presented in *Chapter 6*. The appendices that follow help to understand and validate several parts of the derivation of the algorithms.



# Chapter 2

## Fundamentals

In this chapter, the theory background necessary to understand more deeply this thesis dissertation is presented. Firstly, the reference frames required along this report are presented in Section 2.1. Then, in Section 2.2 the characteristics of the Global Positioning System and the Inertial Navigation System are discussed respectively in Sections 2.2 and 2.3. Finally, in Section 2.4, additional aircraft sensors, such as airspeed and magnetometer sensors, are presented.

### 2.1 Reference Frames

A reference frame is a coordinate system or set of axes within which the position, orientation, and other properties of objects can be measured and analyzed (J. Mulder, Staveren, & Vaart, 2011). In this chapter, different reference frames will be described. The need for various reference frames arises because sometimes the definition of a vector in one reference frame makes more sense and is better understood than in another. Furthermore, the use of difference reference frames will make the derivation of the Equations of Motion (EOM) easier.

Two kinds of reference frames (Earth-fixed and vehicle-fixed), in a total of four, are going to be defined. The only Earth-fixed reference frame is presented in Section 2.1.1 and the three vehicle-fixed reference frames in Sections 2.1.2 to 2.1.4. The transformation between the described reference frames is going to be addressed in Section 2.1.5.

#### 2.1.1 Earth-centered, Earth-fixed reference frame

The first of the reference frames presented is the Earth-centered, Earth-fixed (ECEF) reference frame  $\mathcal{F}_E$  ( $OX_EY_EQ$ ). This is an Earth-fixed frame, as can be seen in Figure 2.1. Its origin  $O$  is located at the center of mass of the Earth,  $X_E$ -axis crosses the Greenwich meridian, the  $Z_E$ -axis is directed along the spin-axis of the Earth and the  $Y_E$ -axis is perpendicular to the previous, passing through the equatorial plane. These three axis form a right-handed coordinate system. Since the  $X_E$ -axis is fixed to the Greenwich meridian this reference frame will rotate along the  $Z_E$ -axis with the angular velocity of the Earth.

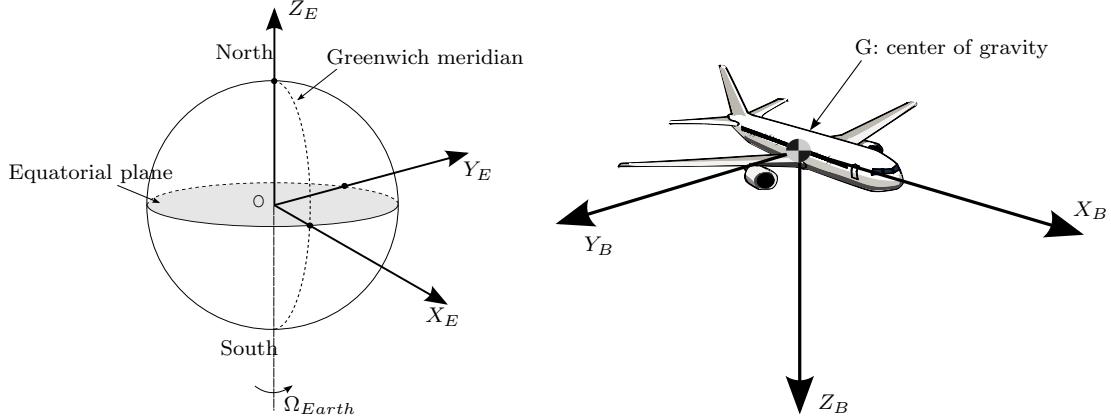
#### 2.1.2 Body-fixed reference frame

The Body-fixed reference frame  $\mathcal{F}_B$  ( $GX_BY_BZ_B$ ) is a vehicle-fixed reference frame as can be seen in Figure 2.2. This is the most common reference frame used by the inertial sensors. Its origin is at the aircraft's center of mass<sup>1</sup>. The  $X_B$ -axis points forward, along the nose of the airplane, in the symmetry

---

<sup>1</sup>If the gravity field is considered constant, the center of mass (*c.m.*) coincides with the center of gravity (*c.g.*) of the aircraft.

plane of the aircraft; the  $Z_B$ -axis also lies in the symmetry plane but points downwards; and finally, the  $Y_B$ -axis is in the direction of the right-wing forming a right-handed orthogonal axis-system.

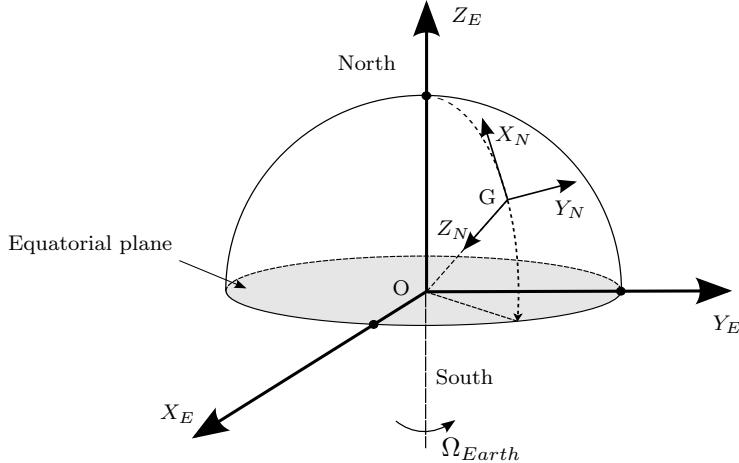


**Figure 2.1:** Earth-centered, Earth-fixed reference frame  $\mathcal{F}_E$ .

**Figure 2.2:** Body-fixed reference frame  $\mathcal{F}_B$ .

### 2.1.3 Navigational reference frame

The Navigational reference frame  $\mathcal{F}_N$  ( $GX_NY_NZ_N$ ) has its origin at the aircraft's center of gravity. It is also known as North-East-Down (NED) reference frame for the reason that the  $X_N$ -axis is directed to the North, the  $Y_N$ -axis to East and  $Z_N$ -axis points to the center of the Earth, i.e., along the gravity direction<sup>2</sup>. The  $X_EY_E$ -plane is often called Local Horizontal plane and follows the variation of the Earth's curvature. The Navigational reference frame is showed in Figure 2.3.



**Figure 2.3:** Navigational reference frame  $\mathcal{F}_N$ .

As we can see from Figure 2.3, in this reference frame the gravity vector  $\underline{g}$  is given by:

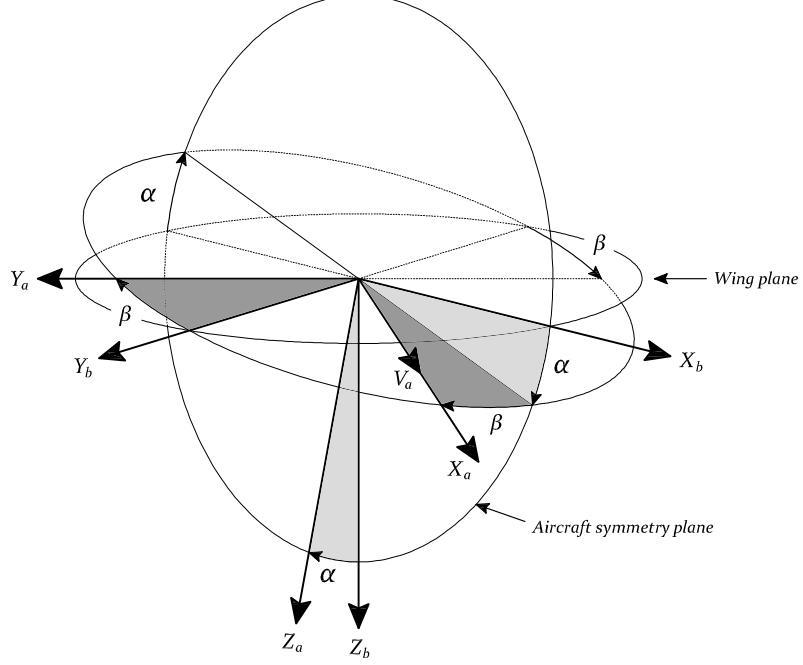
$$\underline{g}_N = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}_N \quad (2.1)$$

---

<sup>2</sup>Note that it is assumed that the Earth is a perfect sphere, otherwise the gravity vector would no longer be directed to the center of the Earth due to its ellipsoid shape.

### 2.1.4 Aerodynamic reference frame

Last but not least, the Aerodynamic reference frame  $\mathcal{F}_A$  ( $GX_A Y_A Z_A$ ) is the third and last vehicle-fixed reference frame presented here (see Figure 2.4). Its origin is also at the center of gravity  $G$  of the aircraft, the  $X_A$ -axis is in the direction of the aerodynamic velocity vector  $\underline{V}_a$ , the  $Z_A$ -axis is in the symmetry plane of the aircraft and the  $Y_A$ -axis is perpendicular to  $GX_A Z_A$  plane to form a right-orthogonal axis system.



**Figure 2.4:** Aerodynamic reference frame  $\mathcal{F}_A$ .

The aerodynamic velocity vector  $\underline{V}_a$  is coupled to the  $\mathcal{F}_A$  reference frame in the following way:

$$\underline{V}_{a_A} = \begin{bmatrix} V_a \\ 0 \\ 0 \end{bmatrix}_A \quad (2.2)$$

However, it is very common to express the aerodynamic velocity vector in the  $\mathcal{F}_B$  reference frame:

$$\underline{V}_{a_B} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}_B \quad (2.3)$$

From Figure 2.4 it is possible to conclude that the orientation of the Aerodynamic reference frame  $\mathcal{F}_A$  with respect to the Body-fixed reference frame  $\mathcal{F}_B$  is related by two angles:

- The *sideslip angle*  $\beta$ : angle between the  $X_A$ -axis and its projection on the plane of symmetry of the airplane.
- The *angle of attack*  $\alpha$ : angle between the  $X_B$ -axis and the  $X_A$ -axis.

They will be further discussed in the next section.

### 2.1.5 Transformations between Reference Frames

As previously said, the use of different reference frames is rather useful. For example, the accelerations are of simpler representation in the Body reference frame and the aircraft position in the ECEF or NED

reference frames. Consequently, different motions are expressed in different frames and in the end it is necessary to get the overall aircraft motion in a particular one. To achieve this, transformation between reference frames are required. These transformations are composed by:

- Translation of origin from the first reference frame to the second reference frame;
- Set of rotations that defines the difference in orientation of two reference frames.

The **Euler Angles** become very handy when rotating reference frames. The orientation from one reference frame to another reference frame can, at any time, be described by three Euler angles:  $[\varphi_1, \varphi_2, \varphi_3]$  (Etkin & Reid, 1996). Commonly, the Euler angles define the attitude of the Body reference frame  $\mathcal{F}_B$ , irrespectively of translation, and there are 12 different ways to perform it.

If the rotation is about the  $X$ -axis by  $\varphi_1$ , then the rotation matrix is given by:

$$R_x(\varphi_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi_1 & \sin \varphi_1 \\ 0 & -\sin \varphi_1 & \cos \varphi_1 \end{bmatrix} \quad (2.4)$$

If the rotation is about the  $Y$ -axis by  $\varphi_2$  the rotation matrix is given by:

$$R_y(\varphi_2) = \begin{bmatrix} \cos \varphi_2 & 0 & -\sin \varphi_2 \\ 0 & 1 & 0 \\ \sin \varphi_2 & 0 & \cos \varphi_2 \end{bmatrix} \quad (2.5)$$

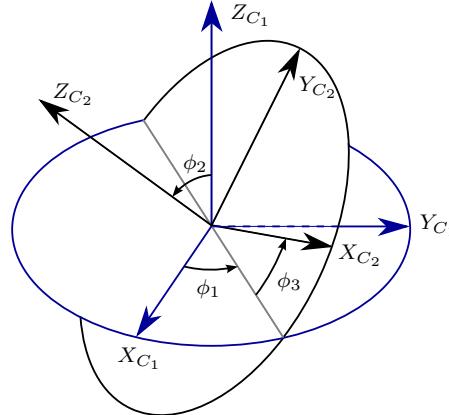
And finally, if the rotation is about the  $Z$ -axis by  $\varphi_3$  the rotation matrix is the following:

$$R_z(\varphi_3) = \begin{bmatrix} \cos \varphi_3 & \sin \varphi_3 & 0 \\ -\sin \varphi_3 & \cos \varphi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Let us consider an example now. In the Figure 2.5, a transformation sequence (3-1-3) is presented. In this case, a vector  $\underline{v}$ , in the reference frame  $C_2$ , is obtained from the reference frame  $C_1$  as follows:

$$\underline{v}_{C_2} = R_z(\varphi_3)R_x(\varphi_2)R_z(\varphi_1)\underline{v}_{C_1} = R_{C_2/C_1}(\varphi_3, \varphi_2, \varphi_1)\underline{v}_{C_1} \quad (2.7)$$

where  $R_{C_2/C_1}$  is the *transformation matrix* that rotates the reference frame  $C_1$  to the reference frame  $C_2$ .



**Figure 2.5:** Example of a transformation 3-1-3, from  $\mathcal{F}_{C_1}$  to  $\mathcal{F}_{C_2}$ .

Since the transformation between reference frames is orthogonal, and after the rotation of a reference frame the vector must keep its original length, the principal characteristics of the transformation matrices are:

1. The determinant of the transformation matrix must be equal to one.
2. The eigenvalues must satisfy one of the orthogonality properties: all eigenvalues are 1 or, one eigenvalue is one and the other two are  $-1$  or, one eigenvalue is 1 and the other two are complex conjugates in the form  $e^{i\theta}$  and  $e^{-i\theta}$ .
3.  $R_{C_1/C_2}^{-1} = R_{C_1/C_2}^T = R_{C_2/C_1}$ .
4.  $R_{C_1/C_2}^T R_{C_1/C_2} = R_{C_1/C_2}^{-1} R_{C_1/C_2} = I$ .

Two transformations will be considered in this thesis report: Navigational reference frame (NED) to Body-fixed reference frame and Body-fixed reference frame to Aerodynamic reference frame, both presented next.

### Navigational reference frame to Body-fixed reference frame

It is now possible to define the transformation from the NED reference frame  $\mathcal{F}_N$  to the Body-fixed reference frame  $\mathcal{F}_B$  presented in Sections 2.1.3 and 2.1.2, respectively. This is one of the most widespread rotations and can be performed in three consecutive rotations:

$$\mathcal{F}_N \rightarrow \mathcal{F}_{N'} \rightarrow \mathcal{F}_{N''} \rightarrow \mathcal{F}_B \quad (2.8)$$

In the aerospace industry the most common rotation sequence is a 3-2-1. For this case we define  $[\varphi_1, \varphi_2, \varphi_3] = [\phi, \theta, \psi]$  and, consequently,  $\psi \rightarrow \theta \rightarrow \phi$ , meaning:

- Rotation around the  $Z_N$ -axis by yaw angle  $\psi$ ;
- Rotation around the  $Y_{N'}$ -axis by pitch angle  $\theta$ ;
- Rotation around the  $X_{N''}$ -axis (which is already  $X_B$ -axis) by roll angle  $\phi$ .

The transformation  $\mathcal{F}_N \rightarrow \mathcal{F}_B$  can now be obtained with the rotation matrices from Eqs. (2.4)- (2.6):

$$\begin{aligned} R_{B/N}(\phi, \theta, \psi) &= R_x(\phi)R_y(\theta)R_z(\psi) \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos \psi \cos \theta & \sin \psi \cos \theta & -\sin \theta \\ \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi & \cos \psi \cos \phi + \sin \psi \sin \theta \sin \phi & \cos \theta \sin \phi \\ \sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi & \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi & \cos \theta \cos \phi \end{bmatrix} \end{aligned} \quad (2.9)$$

### Body-fixed reference frame to Aerodynamic reference frame

In order to perform the transformation  $\mathcal{F}_B \rightarrow \mathcal{F}_A$  two consecutive rotations are required, as can be seen in Figure 2.4. Thus, we will have:

$$\mathcal{F}_B \rightarrow \mathcal{F}_{B'} \rightarrow \mathcal{F}_A \quad (2.10)$$

where  $\mathcal{F}_{B'}$  is the intermediary reference frame. Considering the angle of attack  $\alpha$  positive if  $X_B$  lies above  $X_A$  and the sideslip  $\beta$  positive if  $X_B$  lies on the left of the aerodynamic velocity vector  $\underline{V}_a$ , the sequential rotations to perform  $\mathcal{F}_B \rightarrow \mathcal{F}_A$  are:

- Rotation around  $X_{B'}$ -axis by negative angle of attack angle,  $-\alpha$ .
- Rotation around  $Z_{B'}$ -axis (which is already  $Z_A$ -axis) by sideslip angle  $\beta$ .

And the transformation matrix for  $\mathcal{F}_B \rightarrow \mathcal{F}_A$  can then be obtained:

$$\begin{aligned}
R_{A/B}(\beta, -\alpha) &= R_z(\beta)R_y(-\alpha) \\
&= \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \\
&= \begin{bmatrix} \cos \beta \cos \alpha & \sin \beta & \cos \beta \sin \alpha \\ -\sin \beta \cos \alpha & \cos \beta & -\sin \beta \sin \alpha \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix}
\end{aligned} \tag{2.11}$$

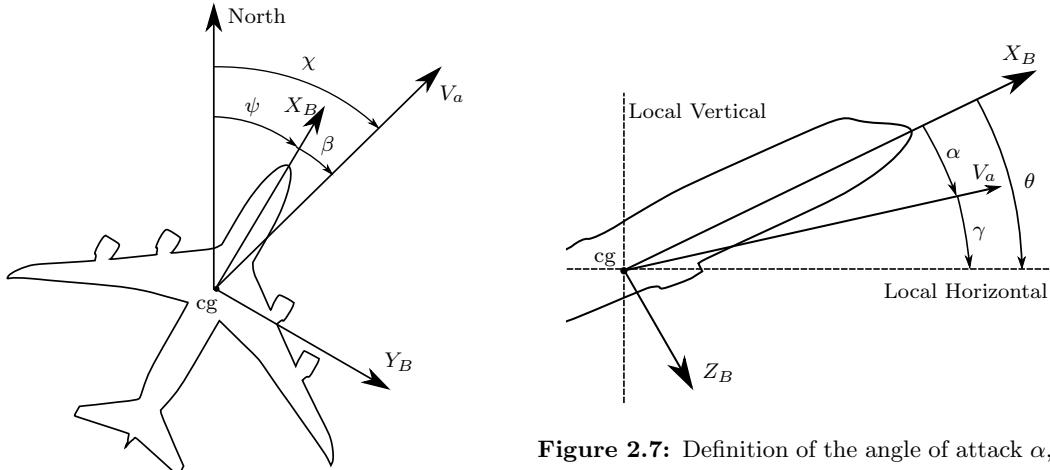
It is rather useful define two additional angles to understand the aircraft orientation. They are the *course angle* and the *flight path angle*:

- **Course Angle:** When the aircraft is flying with non-zero sideslip angle  $\beta$ , the yaw angle  $\psi$  does not correspond to the direction of flight (see Figure 2.6). In this situation, the aircraft flies with an angle called *course*  $\chi$  and has the following relationship with the sideslip  $\beta$  and yaw angle  $\psi$ :

$$\chi = \psi + \beta \tag{2.12}$$

- **Flight Path Angle:** The angle between the airspeed vector ( $X_A$ -axis) and its projection on the horizontal plane ( $X_N Y_N$ -plane) is called *flight path angle*  $\gamma$  (see Figure 2.7). The  $\gamma$  angle is positive if the airplane climbs relative to the air, and negative if the airplane descends. In a symmetric flight<sup>3</sup>, the flight path angle  $\gamma$  is obtained by the difference between the pitch angle  $\theta$  and the angle of attack  $\alpha$  (Ruijgrok, 1996):

$$\gamma = \theta - \alpha \tag{2.13}$$



**Figure 2.6:** Definition of course angle  $\chi$ .

**Figure 2.7:** Definition of the angle of attack  $\alpha$ , climb angle  $\gamma$  and pitch angle  $\theta$ .

---

<sup>3</sup>A symmetric flight is characterized by a zero sideslip angle and the plane of symmetry of the airplane perpendicular to the normal plane of the Earth.

## 2.2 Global Position System

One of the most famous navigational methods at the moment is based in the utilization of satellites, called Global Navigation Satellite System (GNSS). Currently, there are two operational satellite systems, the US Global Positioning System (GPS) and the Russian, GLObal NAVigation Satellite System (GLONASS). Europe is also developing its own GNSS named Galileo. Although there exists two GNSS functioning at the moment, the most used is the GPS (Kaplan & Hegarty, 2006; Nunes, 2011).

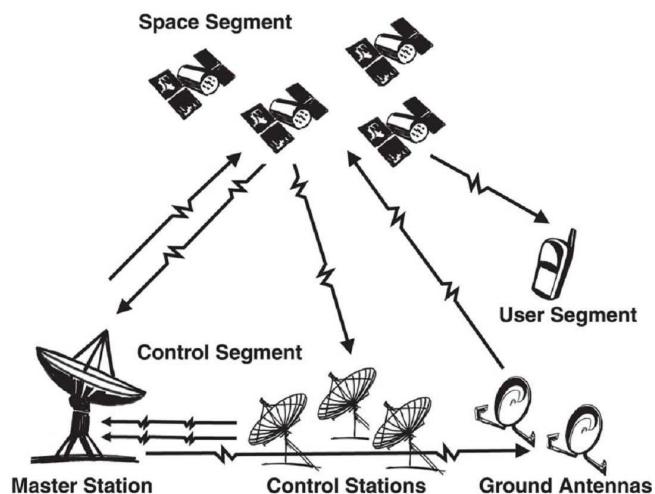
The project to develop GPS started in 1973 so that limitations of previous navigation systems could be overcome. This project was carried by U.S. Department of Defense and became fully operational in 1994, providing global coverage. It was originally run with 24 satellites and only certified military users were able to use the full extend of the system. Nowadays, both military and civilian users can use GPS which became an important role in many research topics regarding navigation. Furthermore, its high price and size were significantly reduced.

GPS applications go far beyond boats and airplanes positioning. Other applications include terrestrial vehicle tracking, cartography, space navigation in Low Earth Orbits, agriculture, roads, high complexity buildings, etc. The GPS is very well described in standard textbooks, such as (El-Rabbany, 2002), and will not be discussed in deep detail here. In Section 2.2.1 some characteristics of the Global Positioning System are presented. After that, the GPS velocity and derived-acceleration are briefly explained in Sections 2.2.2 and 2.2.3, respectively, leaving its modeling/implementation for Chapter 4. Finally, the GPS main sources of errors are described in Section 2.2.4.

### 2.2.1 Principle of the Global Positioning System

The main objective of the GPS is the determination, with the highest degree of precision, of the position, velocity and time, based in noisy observations from the satellite segment (Kaplan & Hegarty, 2006). There are three GPS segments as can be seen in Figure 2.8: space segment, control segment and user segment.

The space segment consists of 24 satellites in six orbital planes with an inclination angle of  $55^\circ$  relative to the Earth's equator. Additionally, the satellites are in an orbit of 20200 km average altitude with a period of 11h58min. The control segment has the objective to monitor the health of the orbiting satellites and uploads navigation data. It consists of various tracking stations located around the world. Finally, the user segment consists of receivers specifically designed to receive, decode and process the GPS satellite signals.



**Figure 2.8:** GPS segments: Space, Control and User (source: <http://connet.us/gps.htm>).

The GPS signals are transmitted in two carrier frequencies: the primary L1 ( $f_{L1} = 1575.42$  MHz) and the secondary L2 ( $f_{L2} = 1227.6$  MHz). The modulation of these signals is performed by the navigation message and by spread spectrum codes with an unique pseudo-random noise sequence for each satellite. The principal carrier frequency L1 is modulated by two codes, namely, the Coarse-Acquisition (C/A) code and the Precision (P) code. The secondary carrier frequency L2 is only modulated by the Precision code. The Precision code is only available to authorized users and make use of the Y-code as encryption, a practice known as anti-spoofing.

Theoretically, three satellites are needed to obtain an unambiguous position fix but, due to the receiver clock quality, a fourth satellite is added to obtain a good position fix correcting the clock error of the user. Measuring the time-delay that the satellite signal needs to reach the user and multiplying by the speed of light yields the spherical range. Eq. (2.14) presents how to calculate the *pseudo-range* ( $PR$ ) of satellite  $i$ .

$$PR_i = c\tau = R_i + c\Delta t_u + \epsilon_i \quad (2.14)$$

where the spherical range  $R_i$  is given by

$$R_i = \sqrt{(x_{si} - x_u)^2 + (y_{si} - y_u)^2 + (z_{si} - z_u)^2} \quad (2.15)$$

The term  $c$  represents the speed of light and  $\Delta t_u$  the clock error of the user. The subscript  $u$  and  $s$  mean, respectively, that the corresponding parameter is relative to the user or to the satellite. Furthermore, additional sources of error exist and have to be taken in consideration upon the receiver positioning/velocity determination. The main source of errors are described in Section 2.2.4 and they are represented in Eq. (2.14) by  $\epsilon$  concerning the sum of various measurement errors of satellite  $i$ : signal propagation, receiver-related and satellite-related.

Aircraft usually fly in open-air and the GPS signal is commonly available. This additional navigational sensor can and should be used due to its characteristics, such as the drift-free feature. More information about the GPS advantages/disadvantages and its relation with this thesis will be presented in Chapter 3.

### 2.2.2 GPS Velocity

The GPS velocity is obtained measuring the *Doppler shift* between the known satellite carrier frequency and the frequency determined at the receiver (Kaplan & Hegarty, 2006). This Doppler shift is directly proportional to the velocity of the receiver, along the direction to the satellite, regardless of the distance to this satellite (Chalko, 2007). It is given by:

$$\Delta f = f_r - f_s = \frac{\dot{PR}_i f_s}{c} \quad (2.16)$$

where  $f_r$  is the frequency of the signal measured at the receiver, and  $f_s$  the frequency of the satellite. Then, the GPS velocity or *range rate* can be calculated by the following equation (M. Mulder, 2010):

$$\dot{PR}_i = \frac{(\underline{x}_{si} - \underline{x}_u)}{R_i} \cdot (\underline{v}_{si} - \underline{v}_u) + c\dot{\Delta}t_u \quad (2.17)$$

In the previous equation,  $v_{si}$  and  $v_u$  are, respectively, the velocity of the satellite  $i$  and the velocity of the user. The accuracy of the Doppler speed measurement varies with the number of tracked satellites as well as on their geometrical distribution above the horizon. Additionally, the satellite velocity has to be precisely known. Nevertheless, the accuracy of the GPS velocity is usually much higher when compared to the GPS position, reaching values better than 1cm/s (Serrano, Kim, & Langley, 2004). Additionally, the GPS provides a drift-free reference vector for the course orientation of the airplane. This characteristic can be used to correct bias present in the inertial sensors.

### 2.2.3 GPS Acceleration

Acceleration determination is not yet a typical application of the GPS. Nevertheless, proposed methods for GPS acceleration determination fall in two categories (Psiaki, Powell, & Kintner, 1999; Zhang, 2005): one is to derive acceleration directly from GPS determined positions (double differentiation) and the other is based on the Doppler shift method differentiation. The latter has several advantages: it does not rely on the precision of the positions from GPS and the accuracy will not dramatically degrade with an increase in the sampling rate (e.g., 10Hz or more). This increase in sampling rate is rather important because one of the biggest disadvantages of the GPS is the low frequency measurements.

As will be seen in Chapter 4, GPS-derived accelerations will be used as an absolute attitude reference by using them in conjunction with a 3-axis accelerometer. This technique was already implemented and results show that a GPS receiver can be used to determine the acceleration of a moving vehicle with reasonable accuracy (Psiaki et al., 1999; Psiaki, 2006).

It has been demonstrated that the GPS is able to provide a measurement of vehicle speed that is sufficiently reliable to determine acceleration with an uncertainty under  $0.10\text{m/s}^2$  (Pinder, Crowe, & Nikiforuk, 2001). Furthermore, a more recent research showed that the accuracy of the solution can reach better results than  $0.01\text{m/s}^2$  (Y. Li et al., 2006). Advantages in using GPS-derived accelerations over the more conventional sensor, the accelerometer, arise with these promising results. They include less susceptibilities to the gravity vector, immunity to vibrational disturbances and temperature fluctuations, and unbiased solutions.

### 2.2.4 GPS Sources of Error

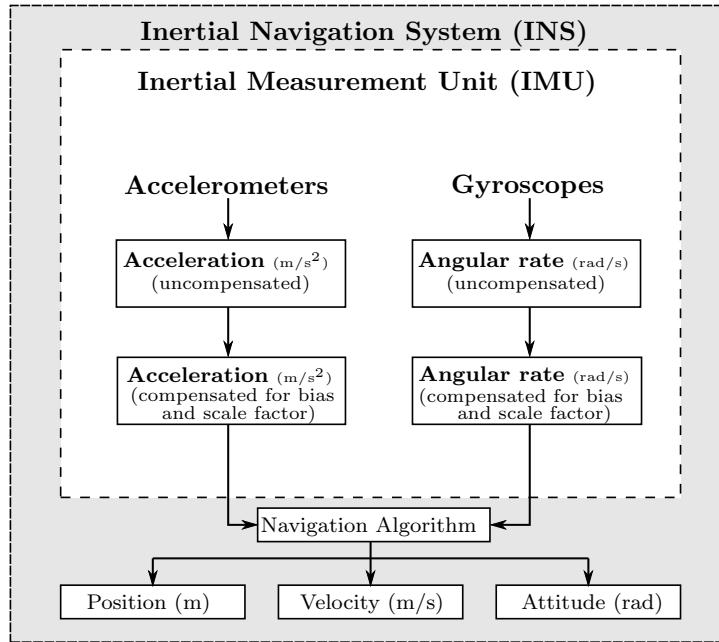
The GPS signal is corrupted by various types of errors before arriving at the receiver unit. A briefly description of main ones is the following (Grewal et al., 2001; Solimeno, 2007; Kaplan & Hegarty, 2006):

- *Satellite clock error*: although the atomic clocks available in the satellites are very accurate, errors can be large enough to require correction due to the difficult synchronization between all the satellites.
- *Receiver clock error*: this is the largest source of error because usually they are inexpensive quartz crystal oscillators which are much less accurate than the satellite clocks.
- *Ionospheric delay*: the presence of free electrons in the upper atmosphere (around 40~1000 km) influence the electromagnetic wave propagation. It is related to the solar radiation and can be estimated by observing the time of arrival of two signals with different frequencies.
- *Tropospheric delay*: this low layer of the atmosphere (0~50 km) is rich in water vapor that refracts the GPS signal leading to a speed reduction (phase delay). It is also related to the satellite elevation.
- *Multipath errors*: besides the direct line of sight of the GPS signal, other reflected signals might also arrive at the receiver distorting the original one. A receiver with a cutoff angle usually helps to reduce this source of error.
- *Satellite ephemeris errors*: differences in the theoretical (estimated) and real satellite orbits results in ephemeris errors that compromise accuracy of the GPS receiver.
- *Measurement error*: the antenna, amplifiers and cables are not perfect and may introduce some noise. Furthermore, the signal quantization noise is commonly present.

## 2.3 Inertial Navigation

Dead-reckoning computations were the heart of every automatic navigation system long before the introduction of the GPS (Barbour, 2004). Its main objective is to obtain the position from Inertial Measurement Units (IMU) such as gyroscopes and accelerometers. With these two inertial sensors, changes in direction (attitude) and position can be measured. The IMU measurements do not require man-made external source (e.g., radio transmitter) and are immune to jamming. However, since integration is required to obtain the position, this will result in accumulated errors. Additionally, the initial position has to be known with a good accuracy.

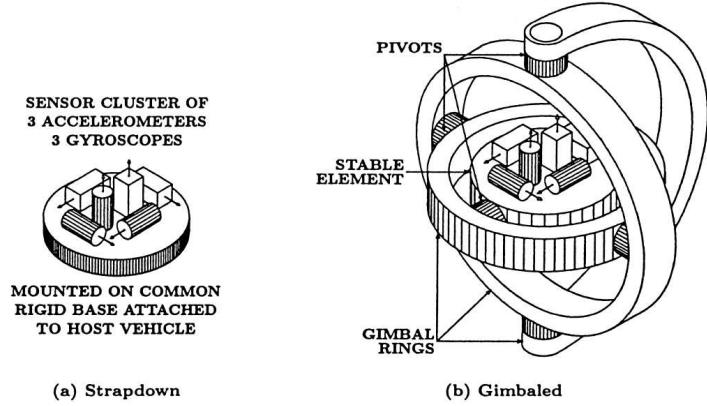
When we talk about a self-contained system that includes the gyroscopes and accelerometers and continuously measures three orthogonal linear accelerations and three angular rates to obtain the object's position, velocity and attitude we are talking about an Inertial Navigation System (INS). The Figure 2.9 illustrates the concept of an INS.



**Figure 2.9:** Concepts of INS and IMU (adapted from (Schultz, 2006)).

There are two different types of INS currently in use (Solimeno, 2007; M. Mulder, 2010; Grewal et al., 2001):

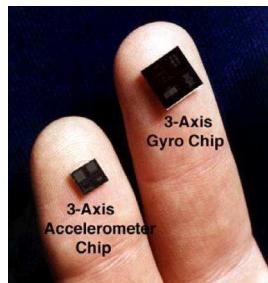
- *The Gimbaled system:* the angular motions of the vehicle are not sensed by the inertial sensors because they are mounted on an isolated gimballed platform (Figure 2.10-b)). This platform maintains a fixed orientation with respect to the Earth. The accelerometers are rigidly attached to the platform and measure the specific force in the navigation frame. The attitude is determined by measuring the relative angles between the vehicle and the platform axes. These systems are very accurate because the sensors can be designed for very small measurements in a small measurements range. However, they are very expensive and rather complex from a mechanical point of view.
- *The Strapdown system:* the inertial sensors are directly attached to the vehicle frame (Figure 2.10-a)). This way, the sensors experience the full dynamic motion of the vehicle. The relationship between the Body-fixed frame and the Navigational frame must be maintained computationally on-board. The mechanical complexity diminishes resulting in a lighter, small and much cheaper system. On the other hand, the accuracy is lower and the computational effort is relatively higher. This is the system considered in this thesis.



**Figure 2.10:** Inertial Gimbal and Strapdown systems, from (Grewal et al., 2001).

### 2.3.1 Low Cost MEMS-based IMUs

Recent developments in micromachining has enabled size and cost reduction of the inertial sensors (see Figure 2.11 from (Barbour, 2004)). This Micro-Electro-Mechanical Systems (MEMS) technology is expected to enable various emerging military and commercial applications (Barbour, 2004). Apart from size and cost reduction, MEMS technology offers other benefits such as production in large scale, power reduction and design flexibility. Additionally, the high working frequency (around 100-200Hz) is a key characteristic when we perform the GPS/INS integration.



**Figure 2.11:** MEMS technology: accelerometer and gyroscope, from (Barbour, 2004).

However, in general, as size decreases, the sensitivity also decreases and the noise increases. The MEMS sensors are more sensitive to the surrounding environment, such as temperature, pressure, magnetic field, etc. All the errors that arise from these sources must be identified, modeled and determined in order to avoid the MEMS's performance to degrade.

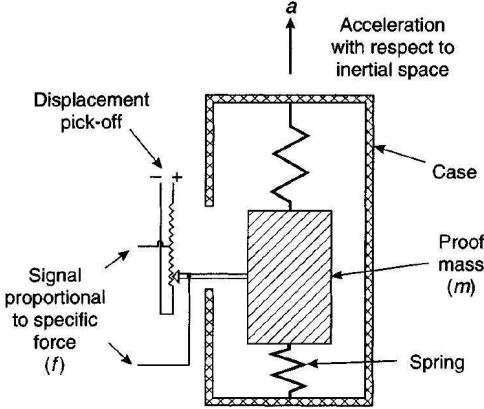
It is assumed that MEMS sensors are used in this thesis and a brief description of the two inertial sensors, accelerometers and gyroscopes, is presented next.

#### Accelerometers

The accelerometers are devices based on Newton's second law and are capable to determine the three-dimensional acceleration vector. Probably, the best way to understand the functioning of a (mechanical) accelerometer is to think that it consists of a proof mass attached by a suspension system to a reference frame; an inertial force will deflect the proof mass; by measuring the magnitude of the force (directly or indirectly, such as through the mass displacement) it is possible to calculate the acceleration (see Figure 2.12). Normally, the accelerometer measures the acceleration in [ $\text{m/s}^2$ ].

Different types of accelerometers have been developed since the first accelerometer, invented by the English physicist George Atwood (1746-1807) in 1783 (Schultz, 2006). It was originally known as the

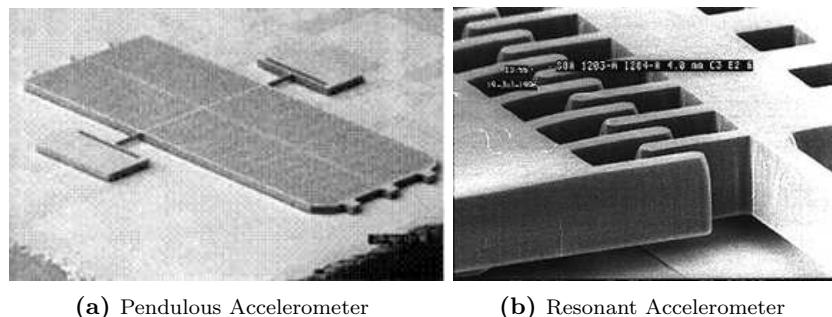
Atwood machine. Generally, there are two main types known as: the *open loop* and *closed loop* accelerometer. The difference is in the way they measure the specific force. The open loop measures the proof mass displacement from the equilibrium position while the closed loop measures the force needed to keep the proof mass in its equilibrium position.



**Figure 2.12:** Conventional mechanical accelerometer, from (Titterton & Weston, 2004).

Currently, the accelerometers are used in a wide variety of scientific and engineering systems. More specifically, the MEMS accelerometers can be less than one mm in each dimension and weigh less than one gram. This new sort of devices detect acceleration in two primary ways (Barbour, 2004):

- The displacement of a hinged or flexure-supported proof mass under acceleration results in a change in a capacitive or piezoelectric readout (*Pendulous Accelerometers*, Figure 2.13a). This type of accelerometers can meet a wide performance range from  $1mg$  for tactical systems down to aircraft navigation quality  $25\mu g$ .
- The change in frequency of a vibrating element is caused by a change in the element's tension induced by a change of loading from a proof mass (*Resonant Accelerometers* or *Vibrating Beam Accelerometers*, Figure 2.13b). This type of accelerometers have the potential for higher performance, down to  $1\mu g$ .

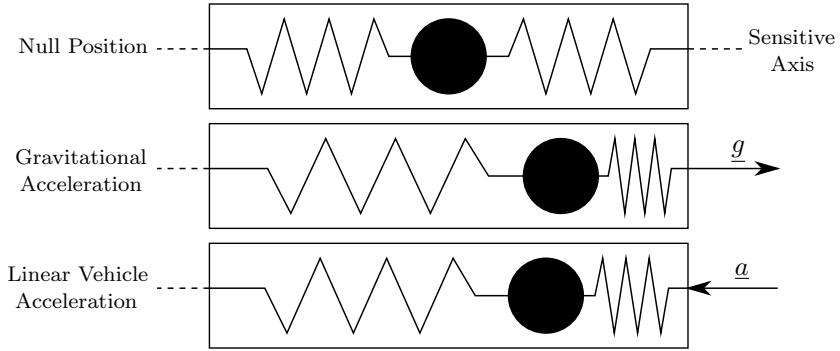


**(a)** Pendulous Accelerometer      **(b)** Resonant Accelerometer

**Figure 2.13:** Types of MEMS accelerometers from (Barbour, 2004).

Finishing this section, it is relevant to describe the measured components of an accelerometer. It measures all types of acceleration irrespective of it comes from gravitational acceleration or from vehicle acceleration. The Figure 2.14 shows an accelerometer in a gravitational field with acceleration in the negative direction of the sensitive axis.

The acceleration is going to be used in inertial navigation, therefore, the acceleration with respect to the inertial frame is needed. The output of an accelerometer is a measurement of the difference between



**Figure 2.14:** Difference between gravitational acceleration and vehicle acceleration, adapted from (Schultz, 2006).

the true vehicle acceleration  $\underline{a}$  and the gravitational acceleration  $\underline{g}$  (Schultz, 2006; M. Mulder, 2010; Psiaki et al., 1999):

$$\underline{f} = \underline{a} - \underline{g} \quad (2.18)$$

where  $\underline{f}$  is the specific force. It is fundamental to know the magnitude and orientation of the gravity vector  $\underline{g}$  with respect to the accelerometer input axis in order to compute the acceleration components of the UAV from the accelerometer outputs.

## Gyroscopes

A gyroscope (often shorten to gyro) is a device to measure or maintain orientation, traditionally based on the principle of conservation of angular momentum, and it was first invented in 1852 by the French physicist Leon Foucault (1819-1868) (Grewal et al., 2001). Although a great variety of technologies and implementations exist, the conventional mechanical gyroscope can be seen as a rigid body similar to a wheel with one principal moment of inertia larger than the other two (see Figure 2.15). The wheel spins about the axis of maximum inertia and it responds to a force applied about the input axis by a reaction force about the output axis. Normally, three gyroscopes are mounted in three distinct orthogonal axes measuring the angular velocity in [rad/s].

Optical gyroscopes (Ring Laser Gyros and Fiber-Optic Gyros) without moving parts have been invented to prevent friction errors and to minimize the size and cost of gyroscopes (Barbour, 2004; Schultz, 2006). The precision of the optical gyroscopes is still not at the same high level as the best mechanical gyros but they can already replace a lot of the common applications for gyroscopes. The principle of the Fiber-Optic gyroscopes is to measure the phase shift between two light beams sent at the same time from the same source. They both go through the same closed fiber optic cable but respectively clockwise and counter-clockwise. If the Fiber-Optic Gyroscope is providing a rotation the two light beams will reach the source at different times due to an increase/decrease of the trajectory of the beams. Then, it results in a phase shift that can be converted to angular velocity. Figure 2.16 shows an example of a modern Optical gyroscope that is sold nowadays. Similarly, the Ring Laser Gyros send two laser beams against three/four mirrors to measure the differences at arrival time.

Advances in MEMS technology and processes have also led to low-cost, high-performance MEMS gyroscopes with lower power consumption and even smaller sizes. Fundamentally, the MEMS gyros fall into three major areas: *Vibrating Beams (Tuning Forks)*, *Vibrating Plates* and *Ring Resonators*. They use the Coriolis acceleration effect on a vibrating proof mass(es) to detect inertial angular rotation (Titterton & Weston, 2004). Thus, these sensors rely on the detection of the force acting on a mass that is subject to linear vibratory motion in a frame of reference which is rotating about an axis perpendicular to both the axis of vibration and the axis about which the rotation is applied.

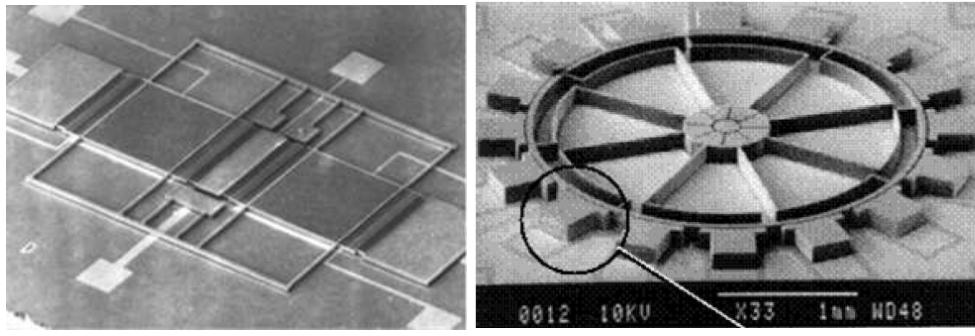


**Figure 2.15:** Conventional mechanical gyroscope  
(source: [www.gyroscopes.org](http://www.gyroscopes.org)).



**Figure 2.16:** Fiber Optic Gyroscope (source: [www.alcielo.com](http://www.alcielo.com)).

However, for inertial MEMS systems, attaining suitable gyros performance is more difficult to achieve than accelerometers performance. Nevertheless, MEMS gyroscopes have been successfully implemented in UAVs or even MAV, (Xu, Zhaoying, Wei, & Qi, 2008; Cheng et al., 2008). Two examples of MEMS gyroscopes are present in Figure 2.17.



(a) Vibrating Plate Gyroscope

(b) Vibrating Ring Gyroscope

**Figure 2.17:** Two types of MEMS gyroscopes from (Barbour, 2004).

MEMS gyroscopes are calibrated during the characterization and qualification process. They do not require re-calibration for most applications. However, for complex and demanding applications such as navigation and dead reckoning, re-calibrate the zero-rate level and sensitivity after the gyroscope is mounted is recommended. For a more deep understanding about the MEMS gyroscopes technology the reader should refer to e.g., (Barbour, 2004; Titterton & Weston, 2004).

### 2.3.2 Inertial Sensor Errors

Despite remarkable advances in MEMS technology in cost and size constraints, MEMS-based inertial sensors have inherited the error behavior of conventional inertial sensors (Park, 2004). Gyroscope errors will result in errors in the transformation matrix between Body-fixed and Navigation frames, whereas accelerometer errors will lead to errors in the integrated velocity and position. Since calibration significantly increases the manufacturing cost, low-cost MEMS inertial sensors are not usually fully calibrated by the manufacturer (Shin, 2005). Thus, the sensor errors need to be modeled in the state vector of the navigation filter as will be seen later.

Briefly, the primary sources of error for both gyroscopes and accelerometers sensors include (Solimeno, 2007; Hummelink, 2011; Premerlani & Bizard, 2009; Grewal et al., 2001):

- *Bias*, which is a non-zero sensor output when the input is zero; bias drift after turn-on needs to be conveniently modeled (usually as a random walk or first-order Gauss-Markov process).
- *Scale factor error*, which represents a non-constant sensor gain (or sensitivity) often resulting from aging or manufacturing tolerances; it can vary during the sensor operation, especially for low-cost IMUs.
- *Non-orthogonality errors*, which result from a misalignment of the sensor axes caused by imperfections in the construction of the sensor assembly. Most stand-alone INS implementation include an initial period for alignment of the attitude direction cosines; however, a GPS-aided alignment is also possible, where the axes misalignments are modeled as part of the INS error dynamics equations.
- *Random noise*, which is an additional signal resulting from the sensor itself or other electronic equipment that interfere with the output signal being measured. It is usually modeled as a zero-mean white Gaussian noise.

These errors will be further described during the inertial sensors modeling in Chapter 4.

## 2.4 Additional Aircraft Sensors

In addition to IMUs and the GPS, other sensors exist, such as magnetometers and air data sensors. The magnetometer, as the name suggests, senses the Earth magnetic field and functions as a compass. On the other hand, the air data instruments which include altitude, airspeed, angle of sideslip and angle of attack sensors, vertical speed indicators, air temperature measuring devices, use a combination of pressure and air flow readings to provide this additional aircraft information to the pilot.

In Section 2.4.1 the different ways to determine the altitude of the aircraft (beyond the GPS) are presented. Then, in Section 2.4.2 the airspeed sensor is described, followed by the direct sensor determination of the sideslip angle and angle of attack in Section 2.4.3. Finally, a brief description on the magnetometers and their actual main problems in integration on UAVs is done in Section 2.4.4. Information about other sensors such as vertical speed indicators or air temperature sensors can be found in e.g., (Kayton & Freid, 1997; Pallett, 1996).

### 2.4.1 Altimetry Sensors

The measurement of altitude is called altimetry and an altimetry sensor measure the altitude of an object above a fixed level. The *radio-altimeters* provide information about the altitude of the aircraft with respect to the ground level directly beneath the aircraft. This altitude can be determined based on time or frequency difference of two signals (Kayton & Freid, 1997). On the other hand, the *barometric-altimeters* work on the aneroid barometer principle. They respond to changes in atmospheric pressure (See Figure 2.18), taken from a static pressure meter, indicating the altitude above a standard pressure level (usually Mean Sea Level). Thus, when the atmosphere is not equal to International Standard Atmosphere (ISA) the measured altitude is not equal to the true altitude.

### 2.4.2 Airspeed Sensor

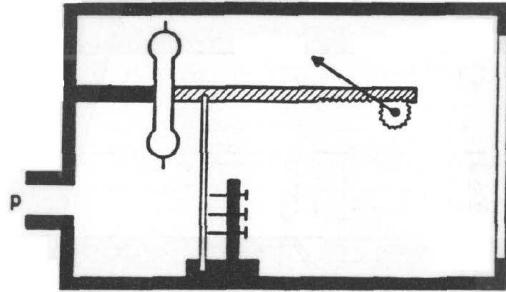
An airspeed sensor measure the True Airspeed (TAS) of the airplane. This is done by taking the difference between the pitot and static pressures detected by either a combined pitot-static probe or, a pitot and static vent separately (Pallett, 1996). Total pressure must be measured at the front opening of the pitot tube (stagnation point) which extends directly into the air flow. Thus, the total pressure is given by the following:

$$p_t = p_s + q \quad (2.19)$$

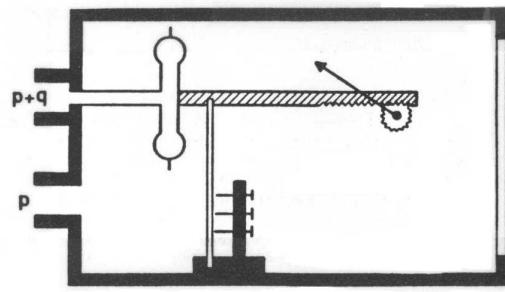
where  $p_t$  is the total pressure,  $p_s$  the static pressure and  $q$  the dynamic pressure, that is given by:

$$q = \frac{1}{2} \rho V_a^2 \quad (2.20)$$

where  $\rho$  is the air density and  $V_a$  the TAS. Due to the usually low velocities of UAVs, the compressibility factors can be disregarded.



**Figure 2.18:** Aneroid static air pressure meter: a vacuum membrane changes its shape as a function of static pressure (M. Mulder, 2010).



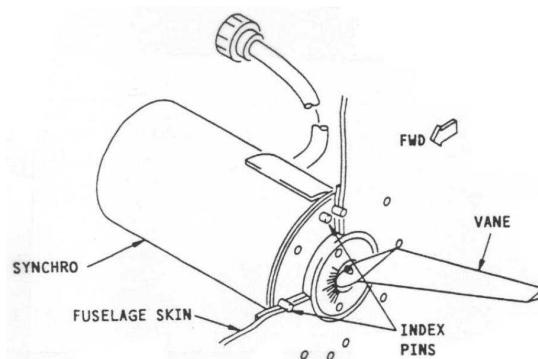
**Figure 2.19:** Pitot-static probe: changes in membrane shape are related to differences between total and static pressure (M. Mulder, 2010).

### 2.4.3 Angle of Attack and Angle of Sideslip Sensors

The angle of attack and angle of sideslip (vanes) sensors are self-aligning and measure the direction of the local flow (Kayton & Freid, 1997). Total pressure must be measured at the front opening on the pitot tube which extends directly into the air flow but at a certain angle with respect to the relative wind. If these angles are kept low (below  $\sim 10$  degrees) the sensors do not produce any significant inaccuracies. Otherwise, special gimbaled pitot tubes might be used. An example of an angle of attack sensor if present in Figure 2.20.

Since the angle of attack is the main variable affecting the aircraft's lift and drag characteristics this sensor is usually included in general aircraft to be primarily used in Stall Warning systems. The sideslip angle sensor is more commonly used in developmental flight test instrumentation and in normal operation it can also be approximated by a body-mounted lateral accelerometer.

As it was said, size, weight and cost are important topics in UAVs design and the free-stream requirement for the sensors implementation could be a difficult aspect to achieve due to the reduced sizes of the UAVs. Additionally, they might be intrusive, disturbing the direction of local flow. Therefore, the integration of these sensors are not common on UAVs.



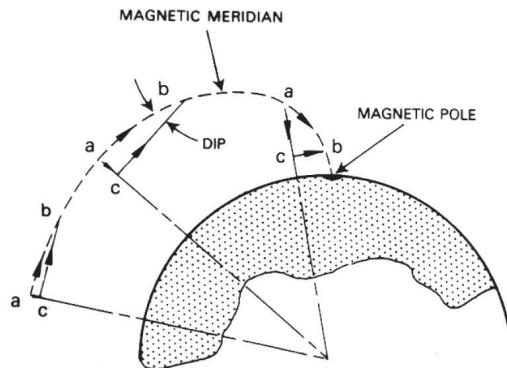
**Figure 2.20:** Angle of attack sensor (Pallett, 1996).

#### 2.4.4 Magnetometers

Magnetometers are used to measure the Earth's magnetic field vector on two or three orthogonal sensors, sometimes in conjunction with a biaxial inclinometer (Persa, 2006). Since this field should point directly North, some method can be used to estimate the heading relative to the magnetic North pole. The horizontal components of the magnetic field are required to compute this heading. If the sensor is tilted with respect to the local horizontal plane, some amount of the vertical component will be sensed by the horizontal axis sensors. For this reason, a two-axis sensing compass can not properly determine the heading of the vehicle if the vehicle is tilted. A sensor with three magnetic sensing axes (3D magnetometer) can determine the orientation of its axes if the magnitude of the magnetic field is known.

A magnetometer updated heading estimate is useful for cases when the sideslip angle needs to be accounted for. Moreover, magnetometer based headings are also helpful in windy situations where the heading obtained from GPS's measurements become inaccurate because the groundspeed becomes small. Such situations have led to vehicles flying uncontrollably downwind for the reason that in the upwind situation they cannot hold a heading estimate sufficient to continue performing the maneuver (Eldredge, 2006). Furthermore the difference between the gyroscopic angles estimation and the magnetometer provided angles holds information about the gyroscopic biases (Hummelink, 2011).

However, the magnetometer is heavily affected by the local magnetic field caused by ferrous metals, batteries and electric motors and its integration in small UAVs with acceptable results has been especially difficult (Cheng et al., 2008; Gebre-Egziabher et al., 2000; Metni, Pfleimlin, Hamel, & Soueeres, 2005). Additionally, a drawback of a GPS/Magnetometer system has to do with the direction of the Earth's magnetic field vector. At the magnetic equator (near the geodetic equator), the Earth's magnetic field vector is horizontal. As one approaches the magnetic poles, the vertical component of magnetic vector (pointing down) tends to dominate (due to the Magnetic Dip, see Figure 2.21). At high latitudes, the magnetic vector and the nominal direction of  $Z_N$  (nadir) might be separated by 20 degrees or less. This short separation degrades the yaw attitude accuracy, and during some maneuvers the two vectors might coincide, making one attitude axis temporarily unmeasurable (Psiaki et al., 1999).



**Figure 2.21:** The Magnetic Dip, from (M. Mulder, 2010). In The Netherlands it is about  $66.7^\circ$ .

Although in the future GPS integrated with other low-cost attitude sensors like magnetometers might lead to a more robust attitude determination system, for aircraft that fly generally in the direction that they are pointed (sideslip angle approximately zero), the GPS measurements are normally enough (Premerlani & Bizard, 2009).



# Chapter 3

## GPS/INS Integration

In this chapter GPS/INS integration topic is presented and described in detail. The advantages of the Satellite and Inertial Navigation systems are discussed and the various types of GPS/INS integration architectures currently available are presented in Section 3.1. Then, one of the most commonly used techniques to estimate the chosen states of a process, the Kalman filtering, is presented in Section 3.2. This *tool* will be posteriorly used in the implementation part of this dissertation.

### 3.1 GPS/INS Integration Architectures

The GPS/INS integration is not a recent topic and several research has already been carried, e.g., (Lewantowicz, 1992; Schmidt, 2004; Titterton & Weston, 2004). This integration has shown to have an overall positive result in the navigation estimation and has the intent of reaching to a navigation system characterized by higher levels of accuracy, reliability and integrity.

In general, an INS exhibits relatively low noise from second to second, but tends to drift over time. On the other hand, GPS errors are relatively noisy from second to second, but exhibit no long-term drift. Integrating the information from each sensor results in a navigation system (single algorithm) that operates like a drift-free INS. There are further benefits added depending on the level at which the information is combined (Schmidt & Phillips, 2003). The Table 3.1 summarizes the main differences between the two navigation systems.

**Table 3.1:** Comparison between Inertial and Satellite Navigation Systems.

	Advantages	Disadvantages
Inertial Navigation Systems	High frequency measurements. Provides both translational and rotational data. Autonomous system. Self-contained (not susceptible to jamming).	Unbounded errors (long term drift). Requires knowledge of gravity.
Satellite Navigation Systems	Errors are bounded.	Low frequency measurements. No (easy) attitude information. Susceptible to jamming.

The main idea of all GPS/INS integration architectures is that the IMUs provide the reference trajectory while the GPS serves as the updating system. This is mainly due to the fact that the INS measurement frequency is much higher than the one from GPS. Differences between the various approaches

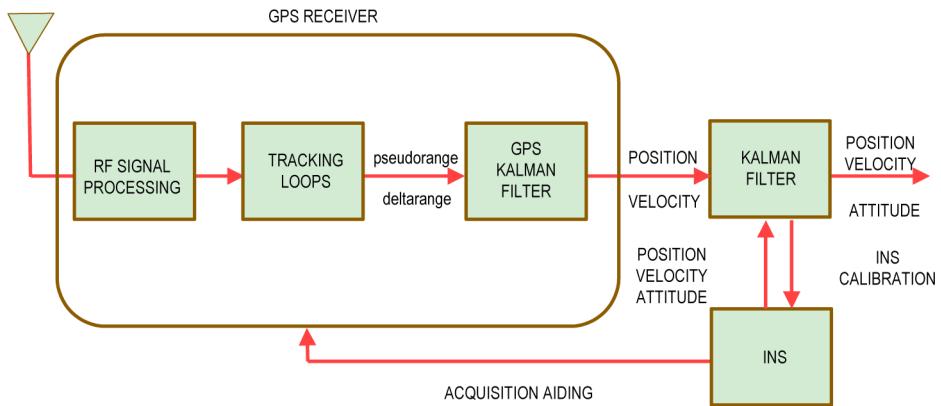
are based on the type of information that is shared between the individual systems. The accuracy of the solution, the resistance to jamming and the ability to calibrate the biases in low-noise Inertial system components depends on the system architecture chosen.

Basically, there are four types of GPS/INS integration architectures (Schmidt & Phillips, 2003): *Uncoupled*, *Loosely coupled*, *Tightly coupled* and *Deeply/Ultra-tight* integrated configurations. The Loosely and Tightly coupled architectures are the most commonly implemented methods and will be described in Section 3.1.1 and Section 3.1.2, respectively. Both strategies can be *open-loop*, where the estimation of the INS errors does not interfere with the operation of the INS, or *closed loop*, where the sensor's errors are compensated within the calculation procedure of the INS mechanization algorithm (Solimeno, 2007). The other two architecture methods will be described in Section 3.1.3. After the description of the architectures, the advantages and disadvantages of each level of integration will be compared in Section 3.1.4.

After the GPS/INS integration architectures presentation, the fundamental notions about Kalman Filtering are presented in Section 3.2. This is the most common technique used to estimate the chosen states of a process and will be posteriorly used in this dissertation.

### 3.1.1 Loosely Coupled Integration

In a *Loosely coupled* integration architecture the two navigation systems operate in cascade, where the GPS provides position and velocity measurements to correct the INS estimations. The GPS keeps its own internal Kalman filter to process the pseudorange and Doppler measurements which are used to obtain the user's position and velocity (Solimeno, 2007). The differences between the INS and GPS calculated positions and velocities are utilized as measurements in a second Kalman filter in which the INS error dynamics equations are used as system model. This way, the second Kalman filter also provides estimates of all the observable INS errors, which are consequently used to correct the INS raw measurements and to compensate the system output. A representative diagram of the Loosely coupled integration architecture is presented in Figure 3.1.



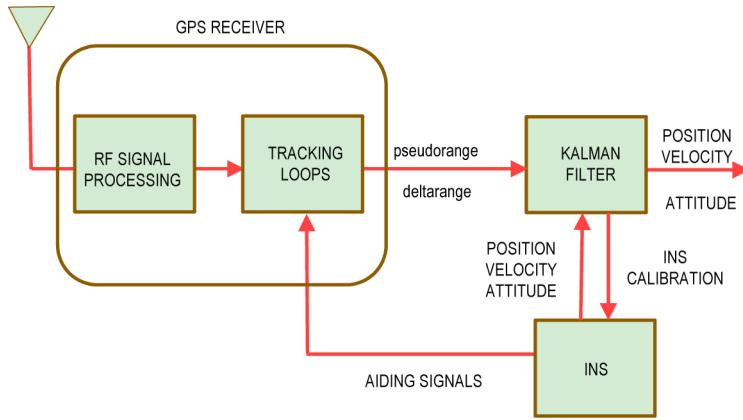
**Figure 3.1:** Loosely coupled GPS/INS integration concept, from (Schmidt, 2004).

The main advantages of this integration scheme are its simplicity in implementation and its robustness. When one of the sensors fail, a solution is still given by the other sensor. On the other hand, one of the disadvantages is the difference in processing time of inertial and satellite navigational systems. When comparing a GPS receiver to a gyroscope or accelerometer, the processing time of the GPS receiver is much larger. This difference causes the GPS receiver output to be delayed (time-latency (Kingston, 2004)), which makes it difficult to integrate the two outputs into one integration filter. Besides the difference in processing time, the sensors provide their output data at different time steps. When it is said that the GPS data rate output is at 1Hz, this is not completely true because the GPS receiver

time processing, to obtain the position and velocity, is dependent on the number of available satellites (Premerlani & Bizard, 2009).

### 3.1.2 Tightly Coupled Integration

In a *Tightly coupled* GPS/INS integration, a single Kalman filter is used. The GPS pseudorange and pseudorange rate measurements are incorporated directly into the navigation estimate, and the position and velocity from the Inertial system are used by the GPS receiver to reduce the tracking loop bandwidths even in the presence of high dynamics (Schmidt & Phillips, 2003; Godha, 2006). A representative diagram of the Tightly coupled integration architecture is presented in Figure 3.2. This GPS/INS integration scheme was successfully implemented with UAV position, velocity, attitude and IMU bias errors estimation e.g., in (George & Sukkarieh, 2005).



**Figure 3.2:** Tightly coupled GPS/INS integration concept, from (Schmidt, 2004).

The biggest advantage is probably that no longer a minimum of four visible satellites is required to obtain a position fix. All the available information is delivered to the filter, even in situations where only one satellite remains observable. Obviously the accuracy will be lower when compared to using four or more satellites but it is a big advantage when compared to Loosely coupled integration.

The disadvantage of this type of architecture is that the state vector increases in size due to the integration of both INS and GPS measurements in a single Kalman filter. This usually leads to an higher processing time (Petovello, 2003).

### 3.1.3 Other Integration Methods

Other GPS/INS integration architectures exist. The most common are the Uncoupled and the Deep/Ultra-Tight integration methods (Schmidt & Phillips, 2003):

- *Uncoupled Integration*: in this architecture the two systems work independently and provide redundancy. The estimated position and velocity measurements from GPS Kalman filter are used to reset the estimated values from Inertial navigation. The system provides no feedback of errors estimate back to the mechanization equation. Consequently, the position and velocity error will keep increasing during the periods that GPS measurements are not available. The advantage of this type of integration is that one of the two can easily be integrated to an existing system due to the non-interaction between the two systems. Additionally, this integration asks for low extra computational power (Hummelink, 2011). However, this method is not very accurate and is not usually used in high-accuracy navigation systems.

- *Deep/Ultra-Tight Integration*: this architecture combines the GPS and the INS in an extremely tight way and it is still under development. A theoretical example can be found e.g., in (D. Li & Wang, 2005; Babu & Wang, 2005). This architecture blends the GPS signal tracking and the GPS/INS integration into a single Kalman filter (Titterton & Weston, 2004). As for the Tightly integration, it does not need a minimum of four satellites. Advantages are related to a higher multi-path resistance, a more effective signal to noise ratio and a faster reacquisition of signal after signal obstruction or jamming. These advantages are achieved at the expense of a greatly increased complexity, computational load and tight time synchronization.

### 3.1.4 Comparison Between the Architectures

It is now time to compare the two most common GPS/INS integration architectures: the Loosely and Tightly coupled integration methods. The specific choice of integration strategy depends on the type of application and operating environment. Their disadvantages and advantages compared to each other can be summarized as:

#### **Loosely Coupled:**

- + Implementation is very simple. There is no need to access and change the GPS and INS hardware.
- + The dimension of the state vectors in both filters are smaller resulting in faster processing times.
- The GPS receiver needs at least four satellites to compute a navigational solution.
- The process noise is added to two Kalman filters instead of only one reducing the capabilities.

#### **Tightly Coupled:**

- + When the number of satellites is below four this architecture still gives an integrated solution.
- + Poor GPS data measurements can be detected and rejected because individual satellite information is used.
- + The noise is added to only one Kalman filter, improving its capabilities.
- Use of a larger size state vector that will require more computational effort (Solimeno, 2007).
- There is a need to access the GPS hardware to obtain the raw data which sometimes might be difficult (Premerlani & Bizard, 2009).

We can conclude that the Tightly coupled integration is better in jamming environments or urban areas. Furthermore, this implementation gives better results when we use low grade, inexpensive MEMS IMUs because of the rapid degradation after loss of lock of the GPS signals (A. Brown & Lu, 2004). However, due to a more complex integration where the access to the GPS hardware is required and the low computational power available on board of small UAVs, the Loosely coupled architecture seems more than adequate. Furthermore, aircraft usually fly in open air and the GPS loss of track is not so common as it happens in land vehicles. Flight test experiences have demonstrated that GPS outages during flights are rare and of a short duration (Gebre-Egziabher et al., 1998). Due to these reasons, the Loosely coupled architecture was chosen as GPS/INS integration architecture in this thesis.

## 3.2 Kalman Filtering

This section gives the fundamental notions about Kalman Filtering. An introduction to the subject is done in Section 3.2.1. Then, the Discrete Kalman filter is presented in Section 3.2.2. After that, some tuning and discretization issues are presented in Sections 3.2.4 and 3.2.5 respectively. The Extended Kalman Filter (EKF), that is going to be implemented in this thesis, is explained in Section 3.2.6. Finally, an explanation on how to test the reliability of the Kalman filter is discussed in Section 3.2.7.

### 3.2.1 Introduction

Kalman filtering was named after Rudolph E. Kalman published in 1960 his paper describing a recursive solution to discrete-data linear filtering problem (Kalman, 1960). In essence, it is a tool that estimates the variables of a wide range of processes, propagating the mean and covariance of the states through time. A set of mathematical equations implement a predictor-corrector type estimator that theoretically minimizes the variance of the estimation error. This peculiarity makes the Kalman Filter (KF) very attractive. Therefore, the practical utility of the filter comes from its ability to estimate a parameter (e.g. Euler angles, aircraft's position, etc.) based on a number of measurements which are (Brogan, 1974):

- Incomplete: related to some but not all of the variables of interest;
- Indirect: related indirectly to the quantities of interest;
- Intermittent: available at irregularly spaced instants of time;
- Inexact: corrupted by various forms of error.

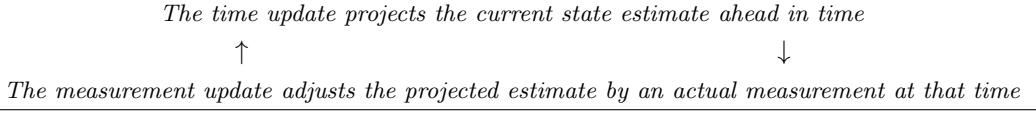
Since its introduction, the Kalman filter has been a subject of extensive research, and many generalizations of it currently exist, including continuous and discrete-time implementations as well as linear and nonlinear variants. All these Kalman filter variations are very well described in several literature such as (Simon, 2006; Grewal et al., 2001) and we will not go into deep detail on the various derivations. Its application is widely spread along the most various technology areas, and navigation industry is one of them. As will be described in the next chapter, the Kalman filter will receive the GPS, gyroscope and accelerometer measurements, filter them and estimate the attitude of an UAV (roll, pitch and yaw angles) and the gyroscope's biases.

The Kalman filter aims to be an extremely effective and versatile procedure to estimate the states of the system, with uncertain dynamics, combining noisy sensor outputs (Grewal et al., 2001). Along this thesis, these parameters are:

- *System states*: it may include position, velocity, acceleration and attitude as well as time-varying parameter of the sensors such as scale factor or output bias. As it will be seen in Chapter 4 the system states will be the attitude parameters of the UAV (roll, pitch and yaw) and biases of the gyroscopes.
- *Noisy sensors*: such as GPS receiver, accelerometers, gyroscopes, airspeed sensors, etc.
- *Uncertain dynamics*: which includes unpredictable disturbances of the UAV, whether caused by human operator or by environmental disturbances.

### 3.2.2 The Discrete Kalman Filter

Having now described the objective of the Kalman filter and its relation with this thesis it is time to present it. In short, the ongoing discrete Kalman filter cycle computation can be described as (Welch & Bishop, 1995):



Consider the general linear, time-varying state-variable model:

$$\dot{\underline{x}}(t) = F(t)\underline{x}(t) + B(t)\underline{u}(t) + G(t)\underline{w}(t), \quad \underline{x}(t_0) = \underline{x}_0 \quad (3.1)$$

and the linear measurements model:

$$\underline{z}(t) = H(t)\underline{x}(t) + D(t)\underline{u}(t) + \underline{v}(t), \quad t = t_i, \quad i = 1, 2, \dots \quad (3.2)$$

where

$\underline{x}(t)$	States vector (dimension $n$ )
$\underline{u}(t)$	Control input vector (dimension $l$ )
$\underline{w}(t)$	System/Process noise vector (dimension $m$ )
$\underline{z}(t)$	Measurements vector (dimension $p$ )
$\underline{v}(t)$	Measurements noise vector (dimension $p$ )
$x_0$	Initial condition of the states vector (dimension $n$ )
$F(t)$	System matrix (dimension $n \times n$ )
$B(t)$	Input matrix (dimension $n \times l$ )
$G(t)$	System noise input matrix (dimension $n \times m$ )
$H(t)$	Observation matrix (dimension $p \times n$ )
$D(t)$	Feedforward matrix (dimension $p \times l$ )

The system noise vector  $\underline{w}(t)$  is a continuous-time white noise process and the measurements noise vector  $\underline{v}(t_i)$  is a discrete-time white noise sequence (the output of the sensors is discrete). Furthermore,  $\underline{w}(t)$  and  $\underline{v}(t_i)$  are mutually uncorrelated for all  $t = t_i, i = 1, 2, \dots$ . Thus, the following holds:

$$E\{\underline{w}(t)\} = 0 \quad (3.3a)$$

$$E\{\underline{w}(t)\underline{w}^T(\tau)\} = Q(t)\delta(t - \tau) \quad (3.3b)$$

$$E\{\underline{v}(t_i)\} = 0 \quad (3.3c)$$

$$E\{\underline{v}(t_i)\underline{v}^T(t_j)\} = R(t_i)\delta_{ij} \quad (3.3d)$$

where  $Q(t)$  and  $R(t_i)$  are, respectively, the *System noise* and *Measurement noise covariance* matrices. The  $\delta(\cdot)$  denotes the Dirac delta function and  $\delta_{ij}$  is the Kronecker delta function<sup>1</sup>. The covariance matrices  $Q$  and  $R$  are positive definite (Grewal & Andrews, 2001).

Since the measurements of the IMUs consist of sampled data, the continuous-time system of Eq. (3.1) has to be transformed into its corresponding discrete-time form:

$$\underline{x}(k+1) = \Phi(k)\underline{x}(k) + \Psi(k)\underline{u}(k) + \underline{w}_d(k) \quad (3.4a)$$

$$\underline{z}(k+1) = H(k+1)\underline{x}(k+1) + D(k+1)\underline{u}(k+1) + \underline{v}(k+1) \quad (3.4b)$$

where

$\Phi(k)$  is the System transition matrix (dimension  $n \times n$ )

$\Psi(k)$  is the Input distribution matrix (dimension  $n \times l$ )

The noise processes  $\underline{w}_d(k)$  and  $\underline{v}(k)$  have the following characteristics:

$$E\{\underline{w}_d(k)\} = 0 \quad (3.5a)$$

$$E\{\underline{w}_d(k)\underline{w}_d^T(j)\} = Q_d(k)\delta_{kj} \quad (3.5b)$$

$$E\{\underline{v}(k)\} = 0 \quad (3.5c)$$

$$E\{\underline{v}(k)\underline{v}^T(j)\} = R(k)\delta_{kj} \quad (3.5d)$$

---

<sup>1</sup>Note that  $\delta_{ij} = 1$  if  $i = j$  and  $\delta_{ij} = 0$  for  $i \neq j$ .

The System noise  $\underline{w}_d(k)$  and Measurement noise  $\underline{v}(k)$  are also assumed uncorrelated:

$$E\{\underline{w}_d(k)\underline{v}^T(j)\} = 0, \quad \forall k, j \quad (3.6)$$

### 3.2.3 The Kalman Filter Formulation

The Kalman filter goal is to estimate the state vector  $\underline{x}$  based on predefined knowledge of the system and observer dynamics and availability of measurements  $\underline{z}$ . It is assumed that:

- The process to be estimated can be modeled in the form specified in Eq. (3.4a).
- The measurements of the system occur at discrete points in time  $t_k = k\Delta t$  (with  $\Delta t$  as sampling time) in accordance with the relationship given in Eq. (3.4b).
- The process and measurement noise,  $\underline{w}_d(k)$  and  $\underline{v}(k)$  respectively, are white with zero-mean and mutually uncorrelated.

Let the *Mean vector*  $\hat{\underline{x}}(k+1|k)$  and the *Error covariance matrix*  $P(k+1|k)$  be defined as:

$$\hat{\underline{x}}(k+1|k) = E\{\underline{x}(k+1)\} \quad (3.7)$$

$$P(k+1|k) = E\{[\underline{x}(k+1) - \hat{\underline{x}}(k+1|k)][\underline{x}(k+1) - \hat{\underline{x}}(k+1|k)]^T\} \quad (3.8)$$

where the notation “ $k+1|k$ ” means a prediction estimate, “ $k+1|k+1$ ” will describe a correction estimate, and  $k$  refers to an iteration of the Kalman Filter. These two different notations represent the two different Kalman filter steps: “ $k+1|k$ ” the Time Update (Prediction), and “ $k+1|k+1$ ” the Measurement Update (Correction). Additionally, the error  $\underline{e}$  between the true state  $\underline{x}$  and the estimate  $\hat{\underline{x}}$  is given by:

$$\underline{e}(k+1|k) = \underline{x}(k+1) - \hat{\underline{x}}(k+1|k) \quad prediction \quad (3.9)$$

$$\underline{e}(k+1|k+1) = \underline{x}(k+1) - \hat{\underline{x}}(k+1|k+1) \quad correction \quad (3.10)$$

The previous implies that the Error covariance matrix of Eq. (3.8) can also be written as:

$$P(k+1|k) = E\{\underline{e}(k+1|k)\underline{e}^T(k+1|k)\} \quad prediction \quad (3.11)$$

$$P(k+1|k+1) = E\{\underline{e}(k+1|k+1)\underline{e}^T(k+1|k+1)\} \quad correction \quad (3.12)$$

Since the objective is to find the state estimate  $\hat{\underline{x}}(k+1|k+1)$  in every iteration, it means that this vector is a linear combination of the prediction estimate  $\hat{\underline{x}}(k+1|k)$  and a weighted difference between the current measurements  $\underline{z}(k+1)$ , and the measurements prediction  $H(k+1)\hat{\underline{x}}(k+1|k)$  in the form (Simon, 2006):

$$\hat{\underline{x}}(k+1|k+1) = \hat{\underline{x}}(k+1|k) + K(k+1)[\underline{z}(k+1) - H(k+1)\hat{\underline{x}}(k+1|k)] \quad (3.13)$$

where  $K(k+1)$  is the weighting factor, known by *Kalman gain*. The term  $[\underline{z}(k+1) - H(k+1)\hat{\underline{x}}(k+1|k)]$  is called the *Measurement Innovation*. This is the part of the measurements that contains new information about the states. The Kalman gain minimizes the correction error covariance  $P(k+1|k+1)$  and it can be computed in the following way (Welch & Bishop, 1995):

1. Substitute Eq. (3.13) in the definition of  $\underline{e}(k+1|k+1)$  in Eq. (3.10).
2. Substitute in the Error covariance matrix in  $P(k+1|k+1)$ , Eq. (3.12).
3. Take the derivative of the result with respect to  $K(k+1)$  and set the result equal to zero.
4. Solve for  $K(k+1)$ .

One of the forms of the Kalman gain  $K(k+1)$  that minimizes Eq. (3.12) is given by,

$$K(k+1) = P(k+1|k)H^T(k+1)[H(k+1)P(k+1|k)H^T(k+1) + R(k+1)]^{-1} \quad (3.14)$$

It is now possible to obtain the error covariance matrix update  $P(k+1|k+1)$  starting from the definition in Eq. (3.12):

$$P(k+1|k+1) = [I - K(k+1)H(k+1)]P(k+1|k) \quad (3.15)$$

Since the previous equation can run into numerical round-off errors and may result in a covariance matrix  $P(k+1|k+1)$  that is no longer symmetrical and semi-positive definite, an alternative to obtain  $P(k+1|k+1)$  is given by (Simon, 2006):

$$\begin{aligned} P(k+1|k+1) &= [I - K(k+1)H(k+1)]P(k+1|k) \\ &\quad [I - K(k+1)H(k+1)]^T + K(k+1)R(k+1)K^T(k+1) \end{aligned} \quad (3.16)$$

The Eqs. (3.13), (3.14) and (3.16) represent the Measurement Update (Correction). Referring now to the Prediction step, the state  $\hat{x}(k+1|k)$  and the covariance matrix  $P(k+1|k)$  are updated according to Eqs. (3.17) and (3.18) (Simon, 2006). They represent the prediction of the state and covariance matrix. The initial state  $\hat{x}_0$  and covariance  $P_0$  are enforced the first time the Time Update (Prediction) is performed.

$$\hat{x}(k+1|k) = \Phi(k)\hat{x}(k|k) + \Psi(k)\underline{u}(k) \quad (3.17)$$

$$P(k+1|k) = \Phi(k)P(k|k)\Phi^T(k) + Q_d(k) \quad (3.18)$$

### 3.2.4 Kalman Filter Tuning Parameters

In order to implement the Kalman filter, the noise statistical information ( $Q$  and  $R$  matrices) needs to be known (Chu, 2011). Both noise covariance matrices should be tuned properly to improve the filter performance. The measurement noise covariance matrix  $R$  is usually measured prior to operation of the filter. It is generally possible to take some off-line sample measurements in order to determine the variance of the measurement noise (Welch & Bishop, 1995). On the other hand, the determination of the process noise covariance matrix  $Q$  is usually more difficult due to the difficulty we have to directly observe the process we are estimating. Hence, a relatively simple process model can produce acceptable results if enough uncertainty is *injected* by choosing high  $Q$  values. Note that the individual process states can be tuned individually by selecting different values on the diagonal of  $Q$ . The off-diagonal elements of  $Q$  represent the process noise coupling effects between the various states.

Under the condition that  $Q$  and  $R$  are constant, both the estimation error covariance  $P(k)$  and the Kalman gain  $K(k)$  will stabilize rather quickly and then remain constant. When this happens, these parameters can be determined by using off-line filtering or for example by determining the steady-state value of  $P(k)$ , (Welch & Bishop, 1995; Grewal & Andrews, 2001).

However, it is frequent that the measurement error does not remain constant. Also, the process noise  $Q(k)$  changes dynamically during filter operation when it is discretized, becoming  $Q_d(k)$ , as will be seen next. The  $Q_d(k)$  might be chosen to account for uncertainty in the model and about the intentions of the user.

### 3.2.5 Kalman Filter Discretization Issues

Most systems in the real world are described with continuous-time dynamics. However, state estimation and control algorithms are almost always implemented in digital computers. This often requires a transformation of continuous-time dynamics to discrete-time dynamics. Thus, it is relevant to refer that the

System transition matrix  $\Phi(k)$  is related to the System matrix  $F(t)$  by (Grewal & Andrews, 2001):

$$\Phi(k) = e^{F(t_k)\Delta t} = \sum_n^{\infty} \frac{(F(t_k)\Delta t)^n}{n!} = I + F(t_k)\Delta t + \frac{(F(t_k)\Delta t)^2}{2!} + \dots \quad (3.19)$$

with  $I$  as Identity matrix. In some cases the high-order terms may be neglected, in order to reduce computation effort. Additionally, the System noise covariance matrix  $Q(t)$  transition to discrete-time  $Q_d(k)$  needs to be addressed. It can be obtained by (Jazwinski, 1970):

$$Q_d(k) = E\{\underline{w}_d(k)\underline{w}_d^T(k)\} = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) G(\tau) Q(\tau) G^T(\tau) \Phi^T(t_{k+1}, \tau) d\tau \quad (3.20)$$

If a trapezoidal integration is applied, the matrix  $Q_d(k)$  is given by (Maybeck, 1994):

$$Q_d(k) \approx \frac{1}{2} [\Phi(k)G(t_k)Q(t_k)G^T(t_k)\Phi^T(k) + G(t_{k+1})Q(t_{k+1})G^T(t_{k+1})] \Delta t \quad (3.21)$$

Finally, the Input distribution matrix is obtained as:

$$\Psi(k) = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) B(\tau) d\tau \quad (3.22)$$

### Kalman Filter Steps Summary

Summarizing each iteration of the Kalman filter, it is composed by the following five operations<sup>2</sup> (for  $k = 1, 2, \dots$ ):

#### Time Update (Prediction)

1. One step ahead state prediction:

$$\hat{x}(k+1|k) = \Phi(k)\underline{x}(k|k) + \Psi(k)\underline{u}(k) \quad (3.23)$$

2. One step ahead error covariance matrix prediction:

$$P(k+1|k) = \Phi(k)P(k|k)\Phi^T(k) + Q_d(k) \quad (3.24)$$

#### Measurement Update (Correction)

3. Kalman gain computation:

$$K(k+1) = P(k+1|k)H^T(k+1)[H(k+1)P(k+1|k)H^T(k+1) + R(k+1)]^{-1} \quad (3.25)$$

4. Update estimate with new measurements:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1)[\underline{z}(k+1) - H(k+1)\hat{x}(k+1|k)] \quad (3.26)$$

5. Update the error covariance matrix:

$$\begin{aligned} P(k+1|k+1) &= [I - K(k+1)H(k+1)]P(k+1|k) \\ &\quad [I - K(k+1)H(k+1)]^T + K(k+1)R(k+1)K^T(k+1) \end{aligned} \quad (3.27)$$

The filter is initialized by assigning values to  $\hat{x}(0|0)$  and  $P(0|0)$ .

---

<sup>2</sup>Note that the Kalman filter algorithm can be written in different forms that leads to the same result.

### 3.2.6 Extended Kalman Filter

As the name states, the EKF is a form of the Kalman Filter “extended” to nonlinear systems. Although it was defined for linear dynamic systems with linear measurements, the Kalman filter has been often used in applications without these characteristics with remarkable success (Grewal et al., 2001). One of the most important features of the EKF is that it uses first-order partial derivatives to linearize nonlinear dynamical systems and/or nonlinear measurements equations at each time-step to estimate the state vector  $\underline{x}$ .

The general nonlinear state-space model is given by (Chu, 2011; Simon, 2006):

$$\dot{\underline{x}}(t) = \underline{f}[\underline{x}(t), \underline{u}(t), t] + G[\underline{x}(t), t] \underline{w}(t), \quad \underline{x}(0) = \underline{x}_0 \quad (3.28)$$

$$\begin{aligned} \underline{z}_m(t) &= \underline{h}[\underline{x}(t), \underline{u}(t), t] \\ \underline{z}(t_k) &= \underline{z}_m(t_k) + \underline{v}(t_k), \quad k = 1, 2, \dots \end{aligned} \quad (3.29)$$

where  $\underline{f}[\cdot]$  is the System (nonlinear) dynamics and  $\underline{h}[\cdot]$  is the Nonlinear observer equation that relates the states to the measurements. For the implementation of the EKF it is assumed that  $\underline{f}[\cdot]$  and  $\underline{h}[\cdot]$  are continuous and continuously differentiable with respect to all elements of  $\underline{x}$  and  $\underline{u}$ . The noise properties of  $\underline{w}(t)$  and  $\underline{v}(t_k)$  are the same as in Eq. (3.3).

The five operations defined in the previous section remain but with some additional changes such as the linearization (with Taylor series) of the nonlinear state and observation Eqs. (3.28)-(3.29) about the nominal values of  $\underline{x}(t)$  and  $\underline{u}(t)$ :  $\underline{x}^*(t)$  and  $\underline{u}^*(t)$ , respectively. The EKF steps are then the following<sup>3</sup>:

#### Time Update (Prediction)

1. *One step ahead state prediction:* for nonlinear systems this can be obtained by integrating the nonlinear state of Eq. (3.28):

$$\hat{\underline{x}}(k+1|k) = \hat{\underline{x}}(k|k) + \int_{t_k}^{t_{k+1}} \underline{f}(\underline{x}(t|t_k), \underline{u}^*(t), t) dt \quad (3.30)$$

2. *One step ahead error covariance matrix prediction:* the covariance matrix is calculated the same way it is for the Kalman filter with the difference of firstly computing the following linearization:

$$F_x(t_k) = \left. \frac{\partial \underline{f}(\underline{x}(t), \underline{u}(t), t)}{\partial \underline{x}(t)} \right|_{\underline{x}(t)=\underline{x}^*(t_k), \underline{u}(t)=\underline{u}^*(t_k), t=t_k} \quad (3.31)$$

where  $\underline{x}^*(t_k) = \hat{\underline{x}}(k|k)$ . Then, after discretization, to get the discrete-time linearized system dynamics  $\Phi(k)$  and the discrete-time system noise covariance matrix  $Q_d(k)$  (e.g., from Eq. (3.21)), the error covariance matrix prediction can be obtained as follows:

$$P(k+1|k) = \Phi(k)P(k|k)\Phi^T(k) + Q_d(k) \quad (3.32)$$

#### Measurement Update (Correction)

3. *Kalman gain computation:* before its calculation, linearization of the observer is necessary:

$$H_x(k+1) = \left. \frac{\partial h(\underline{x}(t), \underline{u}(t), t)}{\partial \underline{x}} \right|_{\underline{x}(t)=\underline{x}^*(t_{k+1}), \underline{u}(t)=\underline{u}^*(t_{k+1}), t=t_{k+1}} \quad (3.33)$$

where  $\underline{x}^*(t_{k+1}) = \hat{\underline{x}}(k+1|k)$ . Then,

$$K(k+1) = P(k+1|k)H_x(k+1)^T [H_x(k+1)P(k+1|k)H_x(k+1)^T + R(k)]^{-1} \quad (3.34)$$

---

<sup>3</sup>These steps are adapted to the implementation in this thesis and other variations exist.

4. Update state estimate with new measurements:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1) [\underline{z}(k+1) - \underline{h}(\hat{x}(k+1|k), \underline{u}^*(k+1))] \quad (3.35)$$

5. Update error covariance matrix:

$$\begin{aligned} P(k+1|k+1) &= [I - K(k+1)H_x(k+1)] P(k+1|k) \\ &\quad [I - K(k+1)H_x(k+1)]^T + K(k+1)R(k+1)K^T(k+1) \end{aligned} \quad (3.36)$$

The EKF is initialized with  $\hat{x}(0|0) = \hat{x}_0$  and  $P(0|0) = P_0$  as the regular Kalman filter. It is important to note that in the EKF implementation the Kalman gain  $K$  calculated in Eq. (3.34) depend on the state estimation through the matrices  $F_x(k)$  and  $H_x(k)$ . Thus, the Kalman gain has to be calculated in real time. This implies that the matrices  $P(k+1|k)$  and  $P(k+1|k+1)$  have to be calculated each step as well. In general, a steady-state solution does not exist for the EKF (Simon, 2006).

### 3.2.7 Reliability Testing

Verifying if the Kalman filter is operating reliably is fundamental. When we run a computer-based simulation using a Kalman filter we can notice if the filter is working properly because it is possible to compare the estimated states with the true states. However, in the real world the real states are not available and it is more difficult to verify that the KF's estimated states are reliable.

It is possible to verify the Kalman filter performance by using the knowledge about the statistics of the *innovations*. The innovation  $\nu$  was defined in Section 3.2.3 as the difference between the actual observation and the predicted observation:

$$\underline{\nu}(k+1) = \underline{z}(k+1) - H(k+1)\hat{x}(k+1|k) \quad (3.37)$$

and it represents the amount of new information introduced in the system by the current measurement. While the Kalman filter is running, it is possible to process the innovations, compute its mean and covariance (Simon, 2006). If the innovation sequence has a zero mean white noise characteristic with covariance given by Eq. (3.38), then the Kalman filter is working properly (Godha, 2006).

$$\sigma_\nu(k+1) = H(k+1)P(k+1|k)H^T(k+1) + R(k+1) \quad (3.38)$$

However, if the innovations sequence is colored, nonzero mean, or has a wrong covariance, there is something wrong with the Kalman filter. The problem with the filter might be that the process model is incorrect or the assumed noise statistics for the process and measurement models are incorrect. Statistical methods can be used to tune  $F(k+1)$ ,  $H(k+1)$ ,  $Q(k+1)$  and  $R(k+1)$  in order to force the innovations to be white zero-mean noise with a covariance given in Eq. (3.38) (Simon, 2006).



# Chapter 4

## Modeling and Implementation

This chapter begins by presenting the modeling of all the sensors used in Section 4.1. Posteriorly, in Section 4.2, two EKF algorithms are developed.

### 4.1 Sensors Modeling

#### 4.1.1 Inertial Measuring Units (IMU) Modeling

In Section 2.3, the main characteristics and error of gyroscopes and accelerometers were presented. Now, these IMU are going to be modeled in order to provide a more reliable and realistic simulation in Chapter 5. First, the errors of the IMU are revised and discussed how they effectively influence the behavior of the gyroscopes and accelerometers. Then, the assembly of all these factors is conducted.

#### IMU Errors Modeling

As previously described in Section 2.3.2, the IMUs are affected by various errors that characterize their performance. In practice, the accelerometers and gyroscopes are affected by biases and time-varying noise (Abdel-Hafez, 2009). These characteristics have to be modeled in order to design a suitable estimation/integration algorithm.

**Measurement Noise:** The *total measurement noise*  $\varepsilon$ , represented in Eq. (4.1), is a combination of white noise  $\varepsilon_w$ , correlated noise  $\varepsilon_c$ , random walk  $\varepsilon_r$ , quantization error  $\varepsilon_q$  and dither noise  $\varepsilon_d$ . The correlated noise results by passing a white noise signal into a memory-filter. The random walk is due to the integration of the white noise signal. The quantization error exists if the outputs of the sensors are digitized before processing. And finally, the dither noise is an intentionally added noise term to randomize quantization errors and noise at discrete frequencies (Hummelink, 2011).

$$\varepsilon = \varepsilon_w + \varepsilon_c + \varepsilon_r + \varepsilon_q + \varepsilon_d \quad (4.1)$$

**Measurement Bias:** The *bias error* of a sensor is composed by a bias at normal operating conditions and an added bias-drift. The latter might be caused by temperature variation. For instance, one of the known reasons why the MEMS gyroscopes lose accuracy very rapidly is because of its large drift (Y. Li et al., 2006). On the other hand, the biases of the accelerometers are commonly negligible for the purpose of estimation methods. Eq. (4.2) defines the measurement's bias, usually present in inertial sensors.

$$\dot{b} = w + \Delta\dot{b}, \quad b(0) = b_0, \quad w \in N(0, b_1^2) \quad (4.2)$$

The  $b$  term is the total bias,  $w$  is a white Gaussian noise that drives the bias to drift (with  $b_1$  as the drift white noise standard deviation). The  $\Delta b$  term represents the bias due to a change in the temperature

of the sensor, usually specified by the manufacturer. The term  $b_1$  is what makes the sensor bias  $b$  time-dependent. It is assumed for simulation purposes that both gyroscopes and accelerometers are calibrated when there is no temperature difference.

**Misalignment Error:** There are two types of misalignment errors: *package misalignment error* and *sensor-to-sensor misalignment error*. The package misalignment error is defined as the angle between the true axis of sensitivity of the sensor and the body axis of the vehicle. On the other hand, the sensor-to-sensor misalignment error defines misalignment error due to non-orthogonality of the IMU axes. In practice, the sensors have small deviations from the defined orientation. Both misalignments errors can be included in the IMU modeling by using a Direction Cosine Matrix that applies three sequential rotations. Performing a 3-2-1 rotation, similar to the one presented in Section 2.1.5, the axis misalignment rotational matrix is given by:

$$R_{mis}(\theta_1, \theta_2, \theta_3) = \begin{bmatrix} \cos \theta_3 \cos \theta_2 & \sin \theta_3 \cos \theta_2 & -\sin \theta_2 \\ \cos \theta_3 \sin \theta_2 \sin \theta_1 - \sin \theta_3 \cos \theta_1 & \cos \theta_3 \cos \theta_1 + \sin \theta_3 \sin \theta_2 \sin \theta_1 & \cos \theta_2 \sin \theta_1 \\ \sin \theta_3 \sin \theta_1 + \cos \theta_3 \sin \theta_2 \cos \theta_1 & \sin \theta_3 \sin \theta_2 \cos \theta_1 - \cos \theta_3 \sin \theta_1 & \cos \theta_2 \cos \theta_1 \end{bmatrix} \quad (4.3)$$

where  $\underline{\theta} = [\theta_1, \theta_2, \theta_3]$  are the misalignment angles for the package or for the sensor-to-sensor misalignments. Defining  $R_p$  as the package misalignment matrix and  $R_s$  as the sensor-to-sensor misalignment matrix, the two matrix can be combined which results in:

$$\underline{A}_m = R_p R_s \underline{A} \quad (4.4)$$

where  $\underline{A}_m$  is the 3-axis measured accelerations vector and  $\underline{A}$  is the 3-axis real accelerations vector. The matrices  $R_p$  and  $R_s$  have the same structure of the matrix in Eq. (4.3). Notice that the Eq. (4.4) also holds for the gyroscopes sensors.

**Scale Factor Error:** The *scale factor error* was defined in Section 2.3.2 as a non-constant sensor gain that often results from aging or manufacturing tolerances. This is usually reported as a nonlinear error function  $f_s$  in the manufacturer specification sheet and is given as a percentage from the measurement full scale.

### Accelerometers and Gyroscopes Final Modeling

It is now possible to summarize the individual errors of the IMUs to derive an accurate description of how (MEMS) accelerometer and gyroscopic sensor data is actually sensed. The measurements of the accelerometers and gyroscopes can be modeled using Eqs. (4.5) and (4.6) respectively. The subscripts  $A$  and  $\omega$  indicate if the parameter is relative to an accelerometer or to a gyroscope respectively;  $\underline{b}$  is the sensor bias computed from Eq. (4.2);  $f_s(\cdot)$  is the scalar factor error function related to each sensor;  $\underline{\varepsilon}$  is the noise of the measurements from Eq. (4.1); and  $R_p$  and  $R_s$  are the misalignment matrices defined in Eqs. (4.3) and (4.4).

$$\underline{A}_m = [I - R_{p_A} R_{s_A}] \underline{A} + \underline{b}_A + f_s(\underline{A}) + \underline{\varepsilon}_A \quad (4.5)$$

$$\underline{\omega}_m = [I - R_{p_\omega} R_{s_\omega}] \underline{\omega} + \underline{b}_\omega + f_s(\underline{\omega}) + \underline{\varepsilon}_\omega \quad (4.6)$$

All the error parameters already described are of high importance to properly model the IMUs. However, manufacturer supplied specification sheets for low-cost MEMS-IMUs rarely provide enough detail to completely construct the error models presented (Xing & Gebre-Egziabher, 2008). Eqs. (4.5) and (4.6) can then be gathered leading to:

$$\begin{aligned} A_{x_m} &= A_x + b_{A_x} + \varepsilon_{A_x} \\ A_{y_m} &= A_y + b_{A_y} + \varepsilon_{A_y} \\ A_{z_m} &= A_z + b_{A_z} + \varepsilon_{A_z} \end{aligned} \quad (4.7)$$

$$\begin{aligned} p_m &= p + b_p + \varepsilon_p \\ q_m &= q + b_q + \varepsilon_q \\ r_m &= r + b_r + \varepsilon_r \end{aligned} \tag{4.8}$$

where the accelerometers vector  $\underline{A}$  and gyroscopes vector  $\underline{\omega}$  were split into their three orthogonal components,  $\underline{A} = [A_x, A_y, A_z]^T$  and  $\underline{\omega} = [p, q, r]^T$ , respectively. The accelerometers and gyroscopes' bias terms  $[b_{A_x}, b_{A_y}, b_{A_z}]$  and  $[b_p, b_q, b_r]$  include all the constant *null-shift* bias and bias-drifts. Additionally, the accelerometers and gyroscopes' noise terms  $[\varepsilon_{A_x}, \varepsilon_{A_y}, \varepsilon_{A_z}]$  and  $[\varepsilon_p, \varepsilon_q, \varepsilon_r]$  are a function of the measurement noise.

The bias of the accelerometers is usually relatively small and can be disregarded, simplifying Eq. (4.7):

$$\begin{aligned} A_{x_m} &= A_x + \varepsilon_{A_x} \\ A_{y_m} &= A_y + \varepsilon_{A_y} \\ A_{z_m} &= A_z + \varepsilon_{A_z} \end{aligned} \tag{4.9}$$

where the  $\varepsilon_A$  terms are modeled as white Gaussian noise<sup>1</sup> frequently specified by the manufacturer.

The bias of the gyroscopes are often modeled as a sum of a *null-shift* bias,  $\underline{b}_0 = [b_{0_p}, b_{0_q}, b_{0_r}]^T$  and a bias-drift  $\underline{b}_R = [b_{R_p}, b_{R_q}, b_{R_r}]^T$ . Thus, from Eq. (4.8) we obtain:

$$\begin{aligned} p_m &= p + b_{0_p} + b_{R_p} + \varepsilon_p \\ q_m &= q + b_{0_q} + b_{R_q} + \varepsilon_q \\ r_m &= r + b_{0_r} + b_{R_r} + \varepsilon_r \end{aligned} \tag{4.10}$$

The *null-shift* bias is modeled as a random constant that varies from turn-on to turn-on:

$$\dot{\underline{b}}_0 = 0 \tag{4.11}$$

The bias-drift, representing the in-run bias variation, is commonly modeled as a first-order Gauss-Markov process (refer to Appendix A for the complete description of this stochastic process). This first-order stochastic differential equation is given by:

$$\dot{\underline{b}}_R = -\frac{1}{\tau_c} \underline{b}_R + \underline{\varepsilon}_{b_R} \tag{4.12}$$

where  $\tau_c$  is the correlation time of the process and  $\underline{\varepsilon}_{b_R}$  is the associated white Gaussian noise. Several methods such as Allan Variance, autocorrelation and power spectrum analysis are available to compute the model parameters (Xing & Gebre-Egziabher, 2008). However, for low-cost MEMS-IMUs these parameters are often not provided by the manufacturer<sup>2</sup>. Thus, the first-order Gauss-Markov process is used to *overbound* the output errors and not necessarily model them exactly. One approach to overcome this lack of sensor information is to use a random walk process instead of a Gauss-Markov process (Machado, 2011):

$$\dot{\underline{b}}_R = \underline{\varepsilon}_{b_R} \tag{4.13}$$

where the noise term  $\underline{\varepsilon}_{b_R}$  incorporates the full noise of the bias-drift and its variance is adjusted manually till reasonable results are obtained.

---

<sup>1</sup>The Appendix A contains a description about all the stochastic processes that are used to model the IMUs.

<sup>2</sup>For instance, the correlation time  $\tau_c$  is not provided by the device used to obtain the UAV's real data.

### 4.1.2 GPS Modeling

After discussion about the Global Positioning System in Section 2.2 the GPS receiver modeling is now addressed. The focus is on the GPS velocity in order to be posteriorly implemented in a GPS/INS integration EKF.

A GPS receiver reports new information as a continuous sequence, typically at 1 to 10Hz rate. However, due to limitations imposed by the GPS's internal signal processing, higher reporting rates do not necessarily mean a better performance (Premerlani & Bizard, 2009). In order to give direction information, the GPS receiver has to move, otherwise there is no way to determine the orientation of the individual GPS antenna. Since the objective is to implement in fixed-wing UAVs, the lack of movement of the GPS receiver should not be a problem. Nevertheless, in case of strong gusts, the UAV's GPS velocity, interpreted as Ground Speed, could be almost zero causing observability problems. However, for the purpose of this thesis the wind is assumed to be zero.

When considering GPS dynamics, another factor has to be kept in consideration: the internal filtering of the GPS receiver. Almost all the GPS units perform some sort of filtering to improve the accuracy of the position and velocity estimation. This filtering will result in a *smoothing effect* on the data when the GPS changes its position or velocity. Instead of presented instantly, the information becomes apparent gradually. Therefore, this *smoothing effect* should be taken into account upon implementation with real GPS data, otherwise a small error would be introduced into navigation calculations during a turn (Premerlani & Bizard, 2009).

The GPS processing time issue is going to be briefly discussed next. After that, the modeling of the GPS velocity and GPS acceleration is presented. These two parameters are going to be used in the GPS/INS integration in the next section.

#### GPS Processing Time

The GPS position and velocity data are not available to the user once the satellite signals arrive at the GPS receiver. All GPS units perform some sort of filtering of the data. The data received from a minimum of four satellites is processed and then transmitted to other systems. The processing time depends on whether the actual position is an *easy* or a *difficult* fix. If a new satellite is flying over, more calculation time will be required as well. There is an inherent compromise in any system between accuracy and transient response (Premerlani & Bizard, 2009). This means that the more accurate a measurement is, the longer it will take to estimate it. A regular GPS processing time is about 0.25 – 0.40s for most GPS receivers with an integrated Kalman filter for signal processing. Upon the integration with the INS, synchronization between the two navigational systems has to be performed. With simulation data the time-latency problem is not placed but it must be addressed upon implementation with real data. The synchronization issue between GPS and IMU measurements will be addressed later.

#### GPS Velocity Modeling

As it was said in Section 2.2.2 the GPS velocity is obtained by measuring the *Doppler shift* between the known satellite carrier frequency and the frequency determined at the receiver. It is assumed that the GPS receiver already provide the velocities in the Navigational reference frame  $\mathcal{F}_N$  (North-East-Down), described in Section 2.1.3. The GPS velocities are defined as:

$$\underline{V}_{GPS} = \begin{bmatrix} u_{GPS} \\ v_{GPS} \\ w_{GPS} \end{bmatrix} \quad (4.14)$$

If the GPS velocities are not in the Navigational reference frame, i.e., if the GPS velocities are in ECEF reference frame for example, a proper transformation  $\mathcal{F}_E \rightarrow \mathcal{F}_N$  can be performed (J. Mulder et al., 2011).

Additionally, for implementation purposes the GPS velocity can be considered an unbiased and drift-free reference vector affected by white Gaussian noise  $\varepsilon$ . The velocities provided by the GPS receiver are given in Eq. (4.15) where the subscript  $m$  represents the output of the receiver. For simulation purposes, the noise will be modeled with a standard deviation  $\sigma$  of 0.01m/s (Chalko, 2007).

$$\begin{aligned} u_{GPSm} &= u_{GPS} + \varepsilon_{u_{GPS}} \\ v_{GPSm} &= v_{GPS} + \varepsilon_{v_{GPS}} \\ w_{GPSm} &= w_{GPS} + \varepsilon_{w_{GPS}} \end{aligned} \quad (4.15)$$

### GPS Acceleration Modeling

The GPS acceleration  $a_{GPS}$  is obtained by differentiating the GPS velocity (*Doppler range-rate*) as described in Section 2.2.3. When performing a differentiation, attention should be paid to implement some kind of filter to avoid excessive noise on the output due to the high-frequency components. Consequently, the differentiated signal will have some phase lag relative to the true acceleration. The GPS-derived acceleration will be modeled using finite differences as presented in Eq. (4.16). Other differentiation methods could be implemented, as the ones presented in (Pavel, 2009), but we would start *losing* track of the highly dynamic characteristics of the UAV. The parameter  $k$  represents the instants when the GPS velocity is available and  $\Delta t$  is the sampling time.

$$\begin{aligned} a_{GPS_x} &= \frac{u_{GPSm}(k) - u_{GPSm}(k-1)}{\Delta t} \\ a_{GPS_y} &= \frac{v_{GPSm}(k) - v_{GPSm}(k-1)}{\Delta t} \\ a_{GPS_z} &= \frac{w_{GPSm}(k) - w_{GPSm}(k-1)}{\Delta t} \end{aligned} \quad (4.16)$$

A compromise should be taken between the sampling time of GPS acceleration and its noise output, i.e., average over long intervals result in velocities that will probably have lower random errors and would give rise to optimistic results when computing acceleration errors (Psiaki et al., 1999).

In order to avoid the problem of noise amplification due to the numerical differentiation of Eq. (4.16), some GPS receivers resort to a PVA (Position, Velocity and Acceleration) dynamics to estimate the acceleration components through the use of an EKF (R. Brown & Hwang, 1997).

## 4.2 Filter Implementation

The filter implementation with GPS/INS integration is going to be treated in this chapter. Here, the most important aspects of this thesis report are presented and then, in Chapter 5, they will be tested. As previously said, the main objective is the attitude determination of a small fixed-wing UAV using reduced computation effort, with the lowest KF complexity, without loss of consistency on the estimation method. Nevertheless, the merits of taking a more general approach must be weighed in terms of computational cost and performance benefit.

Instead of computing always nonlinear numerical solutions, as the Kalman filter's Measurement model for instance, analytical solutions will be attempted in this chapter. Consequently, that will bring us to a *Half* Extended Kalman filter with a nonlinear Dynamics model and a linear Measurement model. Besides the advantage of a reduced computational effort, an exact solution would be attained.

This chapter begins with the derivation of the EOM of the aircraft in Section 4.2.1. After that, in Section 4.2.2, the Process model that is going to be used in the EKF is developed. Then, two different Measurement models will be presented, both making use of analytical solutions that will turn the EKF's measurements observation matrix linear. The main difference between them is that the first model, derived in Section 4.2.3, will make use of GPS-derived accelerations and the second model, presented

in Section 4.2.4, will utilize an airspeed sensor. Additional implementation issues will be discussed in Section 4.2.5.

In the end of the section we will have at our disposal two different EKF to estimate the UAV's attitude:

- **EKF<sub>1</sub>**: with Process model obtained in Section 4.2.2 and Measurement model derived in Section 4.2.3.
- **EKF<sub>2</sub>**: with Process model obtained in Section 4.2.2 and Measurement model derived in Section 4.2.4.

### 4.2.1 Equations of Motion

In this section, the EOM of the aircraft will be derived. This is of high importance for the reason that if we want to describe the dynamics of a vehicle we need a model of it, which is governed by EOM. This set of equations will be obtained under the following assumptions:

- The aircraft is considered a rigid body and its mass is constant.
- The Earth is flat and non-rotating.
- The gravity field is constant.
- The effect of rotating masses is neglected.

In Section 4.2.1 the Translational Equations of Motion are briefly derived. Then, in Section 4.2.1 the Kinematic Differential Equations in terms of Euler angles are developed. These equations will be posteriorly used in the filter implementation.

#### Translational Equations of Motion

First we will derive the Translational Equations of Motion for the aircraft by using the Body-fixed reference frame. Beginning with Newton's second law:

$$\underline{F} = m \frac{d\underline{V}}{dt} = m\underline{a} \quad (4.17)$$

where the assumption *rigid body with constant mass* is taken into consideration. The vector  $\underline{F}$  represents the sum of all external forces applied to the body,  $m$  is its mass and  $\underline{V}$  the linear velocity vector of the center of gravity of the body relative to an inertial reference frame. The acceleration  $\underline{a}$  can be expressed in the Body-fixed reference frame  $\mathcal{F}_B$  as follows (Ruijgrok, 1996):

$$\underline{a} = \frac{d\underline{V}_B}{dt} + \underline{\omega} \times \underline{V}_B = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} qw - rv \\ ru - pw \\ pv - qu \end{bmatrix} \quad (4.18)$$

where  $\underline{\omega} = [p, q, r]^T$  and  $\underline{V}_B = [u, v, w]^T$  were previously defined respectively as the angular velocity of the airplane relative to the inertial non-rotating reference frame and the velocity of the airplane. Consequently, the term  $\frac{d\underline{V}_B}{dt}$  is the time derivative of the velocity vector (with respect to the Body-fixed reference frame).

The external force  $\underline{F}$  is given by the sum of the aerodynamic forces (including the thrust)  $\underline{F}_T$  and the weight of the airplane  $\underline{W}$ . Additionally, in order to represent every term in the Body-fixed reference frame  $\mathcal{F}_B$  we need to rotate the gravity vector  $\mathcal{F}_N \rightarrow \mathcal{F}_B$ :

$$\underline{F} = \underline{F}_T + \underline{W} = \underline{F}_T + R_{B/N}(\phi, \theta, \psi)mg \quad (4.19)$$

with  $R_{B/N}(\phi, \theta, \psi)$  given by Eq. (2.9) and  $\underline{g}$  given by Eq. (2.1). The previous equation becomes:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} F_{T_x} \\ F_{T_y} \\ F_{T_z} \end{bmatrix} + \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} mg \quad (4.20)$$

Combining the Eqs. (4.18) and (4.20) we finally obtain the *Translational Equations of Motion*:

$$\begin{bmatrix} F_{T_x} \\ F_{T_y} \\ F_{T_z} \end{bmatrix} + \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} mg = m \left( \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} qw - rv \\ ru - pw \\ pv - qu \end{bmatrix} \right) \quad (4.21)$$

### Kinematic Differential Equations

Kinematics describes the motion of objects and systems without the consideration of the forces that cause the motion (Wertz, 1978). In this section, the kinematic differential equations of the Body-fixed reference frame  $F_B$  with respect to the Navigational reference frame  $F_N$  are derived. These equations are a set of first-order differential equations that represent the time evolution of the attitude parameters.

As previously defined, in Section 2.1.5, the orientation of the  $F_B$  reference frame relative to  $F_N$  reference frame is described by the Euler angles roll  $\phi$ , pitch  $\theta$  and yaw  $\psi$ . Let us define the time-derivative of the Euler angles, the Euler rates,  $[\dot{\phi}, \dot{\theta}, \dot{\psi}]$ . In order to obtain the relation between these two rotational vectors a sequence of three rotations (chosen as 3-2-1) has to be applied:

$$N \xrightarrow{R_z(\psi)} N' \xrightarrow{R_y(\theta)} N'' \xrightarrow{R_x(\phi)} B$$

Where  $R_x(\phi)$ ,  $R_y(\theta)$  and  $R_z(\psi)$  are the rotation matrices defined in Section 2.1.5, and  $N'$  and  $N''$  the intermediate reference frames. If the additive property of angular velocity is taken into consideration the following relationship between the rotational rates and the Euler rates is obtained (J. Mulder et al., 2011):

$$\begin{aligned} \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_x(\phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_x(\phi)R_y(\theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \end{aligned} \quad (4.22)$$

Multiplying the matrices leads to:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (4.23)$$

By rearranging Eq. (4.23), the *Kinematic Differential Equations* (in terms of Euler Angles) are obtained:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.24)$$

### 4.2.2 Process Model

The nonlinear EOM used to describe the UAV are kinematic in nature. Since the objective is to estimate the UAV's attitude, the roll  $\phi$ , pitch  $\theta$  and yaw  $\psi$  angles are chosen as state variables. Additionally, due

to the previous discussion about the bias(-drift) of the MEMS-gyroscopes, their bias  $\underline{b} = [b_p, b_q, b_r]^T$  are chosen as state variables as well. With this added bias states, the computational effort increases but also the estimation performance. Thus, the final state vector is:

$$\underline{x} = [\phi, \theta, \psi, b_p, b_q, b_r]^T \quad (4.25)$$

and the continuous-time system, that will be derived, is described by (recall Eq. (3.28)):

$$\dot{\underline{x}}(t) = \underline{f}[\underline{x}(t), \underline{u}(t), t] + G[\underline{x}(t), t] \underline{w}(t) \quad (4.26)$$

The rate of change of the states is given in Eq. (4.27) where the Kinematic differential equations from Eq. (4.24) were used. Note that the time dependency term “ $t$ ” was dropped without loss of generality. The three bias states  $[b_p, b_q, b_r]$  were described in Section 4.1.1. For ease of notation, the subscript  $R$  of Eq. (4.12) was dropped.

$$\dot{\underline{x}} = \begin{cases} \dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ \dot{\theta} = q \cos \phi - r \sin \phi \\ \dot{\psi} = q \frac{\sin \phi}{\cos \theta} + r \frac{\cos \phi}{\cos \theta} \\ \dot{b}_p = -\frac{1}{\tau} b_p + \varepsilon_{b_p} \\ \dot{b}_q = -\frac{1}{\tau} b_q + \varepsilon_{b_q} \\ \dot{b}_r = -\frac{1}{\tau} b_r + \varepsilon_{b_r} \end{cases} \quad (4.27)$$

It is presumed that the angular rates  $\underline{\omega} = [p, q, r]^T$  are available, and they form the input vector  $\underline{u}$ . These angular rates were modeled in Eq. (4.8) as:

$$\begin{aligned} p_m &= p + b_p + \varepsilon_p & p &= p_m - b_p - \varepsilon_p \\ q_m &= q + b_q + \varepsilon_q & \Leftrightarrow & q = q_m - b_q - \varepsilon_q \\ r_m &= r + b_r + \varepsilon_r & r &= r_m - b_r - \varepsilon_r \end{aligned} \quad (4.28)$$

If the previous equation is replaced in Eq. (4.27) we get:

$$\dot{\underline{x}} = \begin{cases} \dot{\phi} = p_m - b_p - \varepsilon_p + (q_m - b_q - \varepsilon_q) \sin \phi \tan \theta + (r_m - b_r - \varepsilon_r) \cos \phi \tan \theta \\ \dot{\theta} = (q_m - b_q - \varepsilon_q) \cos \phi - (r_m - b_r - \varepsilon_r) \sin \phi \\ \dot{\psi} = (q_m - b_q - \varepsilon_q) \frac{\sin \phi}{\cos \theta} + (r_m - b_r - \varepsilon_r) \frac{\cos \phi}{\cos \theta} \\ \dot{b}_p = -\frac{1}{\tau} b_p + \varepsilon_{b_p} \\ \dot{b}_q = -\frac{1}{\tau} b_q + \varepsilon_{b_q} \\ \dot{b}_r = -\frac{1}{\tau} b_r + \varepsilon_{b_r} \end{cases} \quad (4.29)$$

and after some manipulation the following is obtained:

$$\dot{\underline{x}} = \begin{cases} \dot{\phi} = p_m - b_p + (q_m - b_q) \sin \phi \tan \theta + (r_m - b_r) \cos \phi \tan \theta - \varepsilon_p - \varepsilon_q \sin \phi \tan \theta - \varepsilon_r \cos \phi \tan \theta \\ \dot{\theta} = (q_m - b_q) \cos \phi - (r_m - b_r) \sin \phi - \varepsilon_q \cos \phi + \varepsilon_r \sin \phi \\ \dot{\psi} = (q_m - b_q) \frac{\sin \phi}{\cos \theta} + (r_m - b_r) \frac{\cos \phi}{\cos \theta} - \varepsilon_q \frac{\sin \phi}{\cos \theta} - \varepsilon_r \frac{\cos \phi}{\cos \theta} \\ \dot{b}_p = -\frac{1}{\tau} b_p + \varepsilon_{b_p} \\ \dot{b}_q = -\frac{1}{\tau} b_q + \varepsilon_{b_q} \\ \dot{b}_r = -\frac{1}{\tau} b_r + \varepsilon_{b_r} \end{cases} \quad (4.30)$$

Comparing Eq. (4.30) with the continuous-time (state-space) system of Eq. (4.26), the following results are obtained:

$$\underline{f}[\underline{x}(t), \underline{u}(t), t] = \begin{bmatrix} p_m - b_p + (q_m - b_q) \sin \phi \tan \theta + (r_m - b_r) \cos \phi \tan \theta \\ (q_m - b_q) \cos \phi - (r_m - b_r) \sin \phi \\ (q_m - b_q) \frac{\sin \phi}{\cos \theta} + (r_m - b_r) \frac{\cos \phi}{\cos \theta} \\ -\frac{1}{\tau} b_p \\ -\frac{1}{\tau} b_q \\ -\frac{1}{\tau} b_r \end{bmatrix} \quad (4.31)$$

$$G[\underline{x}(t), t] = \begin{bmatrix} -1 & -\sin \phi \tan \theta & -\cos \phi \tan \theta & 0 & 0 & 0 \\ 0 & -\cos \phi & \sin \phi & 0 & 0 & 0 \\ 0 & -\frac{\sin \phi}{\cos \theta} & -\frac{\cos \phi}{\cos \theta} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.32)$$

and,

$$\underline{w} = [\varepsilon_p, \varepsilon_q, \varepsilon_r, \varepsilon_{b_p}, \varepsilon_{b_q}, \varepsilon_{b_r}]^T \quad (4.33)$$

This system model will remain unchanged for all the derived Kalman filters. Since this process model is nonlinear and the objective is to implement it in an EKF, the dynamics Eq. (4.26) has to be linearized. From Eq. (3.31), we obtain:

$$\begin{aligned} F_x &= \frac{\partial f(\underline{x}, \underline{u})}{\partial \underline{x}} = \\ &= \begin{bmatrix} (q_m - b_q)c_\phi t_\theta - (r_m - b_r)s_\phi t_\theta & (t_\theta^2 + 1)((q_m - b_q)s_\phi + (r_m - b_r)c_\phi) & 0 & -1 & -s_\phi t_\theta & -c_\phi t_\theta \\ -(q_m - b_q)s_\phi - (r_m - b_r)c_\phi & 0 & 0 & 0 & -c_\phi & s_\phi \\ (q_m - b_q)\frac{c_\phi}{c_\theta} - (r_m - b_r)\frac{s_\phi}{c_\theta} & (q_m - b_q)\frac{s_\phi s_\theta}{c_\theta^2} + (r_m - b_r)\frac{c_\phi s_\theta}{c_\theta^2} & 0 & 0 & -\frac{s_\phi}{c_\theta} & -\frac{c_\phi}{c_\theta} \\ 0 & 0 & 0 & -\frac{1}{\tau} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix} \end{aligned} \quad (4.34)$$

where  $c(\cdot)$ ,  $s(\cdot)$  and  $t(\cdot)$  correspond respectively to  $\cos(\cdot)$ ,  $\sin(\cdot)$  and  $\tan(\cdot)$ .

#### 4.2.3 Measurement Model 1

Numerous different solutions to obtain an attitude measurement have been proposed and implemented. For instance, one common method is to use the carrier phase of GPS signals with three or more GPS antennas placed with a known geometry (Chiang, Wang, Chang, & Peng, 2002). Since the solution improves as the baseline between antennas increases, it makes the implementation on small UAVs/MAVs more difficult. Other methods make use of vector measurements of the magnetic and gravitational fields and then solve a set of nonlinear equations using optimization techniques (Gebre-Egziabher et al., 2000). Another innovative approach was to use the signal-to-noise ratio on the GPS antenna to derive the attitude (Lightsey & Madsen, 2003).

By augmenting the accelerometer with GPS-derived accelerations (recall Eq. (2.18)), a estimation of the roll  $\phi$  and pitch  $\theta$  angles of the aircraft can be made. Additionally, the course<sup>3</sup> of the UAV can be determined by the GPS (North and East) velocities. This implies that the AHRS presented here will only be valid if fixed-wing UAVs, with sufficient GPS velocities to produce an heading measurement, are used. More information about the heading observability and possible improvements are presented in Section 5.1.5.

The measurements vector  $\underline{z}$  will be composed by the three Euler angles, roll  $\phi$ , pitch  $\theta$  and yaw  $\psi$  that will now be derived. From Eq. (2.12) and assuming that the sideslip angle  $\beta \approx 0$ , the yaw angle is given by<sup>4</sup>:

$$\psi = \chi - \beta \approx \chi \quad (4.35)$$

where  $\chi$  was previously defined as the course angle. Thus, with the previous assumption, the measured yaw angle can be obtained with the GPS velocity components as follows:

$$\psi_m = \arctan \left( \frac{v_{GPSm}}{u_{GPSm}} \right) \quad (4.36)$$

<sup>3</sup>That could be coincident to the yaw angle, depending if the sideslip is zero.

<sup>4</sup>A negligible sideslip is often considered in an aircraft with a fixed rudder and reasonable aerodynamic characteristics.

The measurements of the UAV's roll  $\phi_m$  and pitch  $\theta_m$  angles can be obtained by measuring acceleration in the Body-fixed reference frame  $\mathcal{F}_B$  and relating it to the Navigational reference frame  $\mathcal{F}_N$ . Recall from Eq. (2.18) that the accelerometer measures the true vehicle acceleration minus the gravity:

$$\underline{f} = \underline{a} - \underline{g} \quad (4.37)$$

which is similar to the Eq. (4.20). Under the assumption that the GPS velocities are obtained in inertial space and transforming all the components to Body-fixed reference frame, the measurements of the accelerometers are then equal to the difference between the GPS-derived accelerations and the gravity:

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_B = R_{B/N}(\phi, \theta, \psi) \begin{bmatrix} a_{GPS_x} \\ a_{GPS_y} \\ a_{GPS_z} \end{bmatrix}_N - R_{B/N}(\phi, \theta, \psi) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}_N \quad (4.38)$$

where the matrix  $R_{B/N}(\phi, \theta, \psi)$  is given by Eq. (2.9) and the GPS-derived accelerations  $a_{GPS}$  are given by Eq. (4.16). Rewriting Eq. (4.38), a *nonlinear measurement model*, with the accelerations from the accelerometers sensors as measurements, is obtained:

$$\begin{aligned} \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_B &= R_{B/N}(\phi, \theta) \begin{bmatrix} a_{GPS_x} \cos \psi + a_{GPS_y} \sin \psi \\ -a_{GPS_x} \sin \psi + a_{GPS_y} \cos \psi \\ a_{GPS_z} - g \end{bmatrix} \\ &= \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ \sin \phi \sin \theta & \cos \phi & \cos \theta \sin \phi \\ \cos \phi \sin \theta & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \end{aligned} \quad (4.39)$$

where the vector  $[r_x, r_y, r_z]^T$  was defined as:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} a_{GPS_x} \cos \psi + a_{GPS_y} \sin \psi \\ -a_{GPS_x} \sin \psi + a_{GPS_y} \cos \psi \\ a_{GPS_z} - g \end{bmatrix} \quad (4.40)$$

which includes the vehicle's GPS-derived accelerations and gravity acceleration already rotated by the angle  $\psi = \psi_m$ , determined in Eq. (4.36). If the sideslip is not assumed to be zero and there is a way to obtain it, the Eq. (4.40) would become:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} a_{GPS_x} (\cos \chi \cos \beta + \sin \chi \sin \beta) + a_{GPS_y} (\sin \chi \cos \beta - \sin \beta \cos \chi) \\ -a_{GPS_x} (\sin \chi \cos \beta - \sin \beta \cos \chi) + a_{GPS_y} (\cos \chi \cos \beta + \sin \chi \sin \beta) \\ a_{GPS_z} - g \end{bmatrix} \quad (4.41)$$

where the  $\beta$  could be integrated. Nevertheless, continuing with the assumption of Eq. (4.35), the Eq. (4.39) is clearly nonlinear and could be integrated in an EKF as measurement model. However, finding an analytical solution for  $\phi$  and  $\theta$  would allow us to use all the three Euler angles as *measurements*. Thus, solving the first equation of Expression (4.39) for  $\theta$ :

$$\begin{aligned} A_x &= r_x \cos \theta - r_z \sin \theta \\ \Leftrightarrow \frac{A_x}{\cos \theta} &= r_x - r_z \tan \theta \\ \Leftrightarrow \frac{A_x}{\frac{1}{\pm \sqrt{1+\tan^2 \theta}}} &= r_x - r_z \tan \theta \\ \Leftrightarrow (1 + \tan^2 \theta) A_x^2 &= (r_x - r_z \tan \theta)^2 \\ \Leftrightarrow A_x^2 - r_x^2 + A_x^2 \tan^2 \theta - r_z^2 \tan^2 \theta + 2r_x r_z \tan \theta &= 0 \\ \Leftrightarrow (A_x^2 - r_z^2) \tan^2 \theta + 2r_x r_z \tan \theta + A_x^2 - r_x^2 &= 0 \end{aligned} \quad (4.42)$$

where the following trigonometric relation between  $\cos \theta$  and  $\tan \theta$  was applied:

$$\cos \theta = \frac{1}{\pm \sqrt{1 + \tan^2 \theta}} \quad (4.43)$$

Eq. (4.42) is a second-order equation of  $\tan \theta$  that has the following quadratic solution:

$$ax^2 + bx + c = 0 \Leftrightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (4.44)$$

with,

$$\begin{aligned} x &= \tan \theta \\ a &= A_x^2 - r_z^2 \\ b &= 2r_x r_z \\ c &= A_x^2 - r_x^2 \end{aligned}$$

Finally, after some manipulation, the analytical solution for the *measured* pitch angle  $\theta = \theta_m$  can be found:

$$\theta_m = \arctan \left( \frac{-r_x r_z \pm A_x \sqrt{r_x^2 + r_z^2 - A_x^2}}{A_x^2 - r_z^2} \right) \quad (4.45)$$

If the same strategy is taken for the second equation of Expression (4.39) the analytical solution for the *measured* roll angle  $\phi = \phi_m$  can also be found:

$$\begin{aligned} A_y &= r_x \sin \theta \sin \phi + r_y \cos \phi + r_z \cos \theta \sin \phi \\ \Leftrightarrow A_y &= (r_x \sin \theta + r_z \cos \theta) \sin \phi + r_y \cos \phi \\ \Leftrightarrow A_y &= r_\theta \sin \phi + r_y \cos \phi \end{aligned} \quad (4.46)$$

where the following was defined,

$$r_\theta = r_x \sin \theta + r_z \cos \theta \quad (4.47)$$

with  $\theta = \theta_m$ , already determined in Eq. (4.45). Performing similar manipulations as the ones applied to obtain the  $\theta_m$  solution, an analytical solution for the roll angle  $\phi_m$  is given by:

$$\phi_{m_1} = \arctan \left( \frac{r_\theta r_y \pm A_y \sqrt{r_y^2 + r_\theta^2 - A_y^2}}{A_y^2 - r_\theta^2} \right) \quad (4.48)$$

Another solution for  $\phi_m$  but using the  $A_z$  accelerometer can be obtained by solving the third equation of Expression (4.39), leading to:

$$\phi_{m_2} = \arctan \left( \frac{-r_\theta r_y \pm A_z \sqrt{r_\theta^2 + r_y^2 - A_z^2}}{A_z^2 - r_y^2} \right) \quad (4.49)$$

With analytical solutions for the Euler angles, the measurements vector  $\underline{z}$  of the EKF finally become:

$$\underline{z} = \begin{bmatrix} \phi_m \\ \theta_m \\ \psi_m \end{bmatrix} = \begin{bmatrix} \arctan \left( \frac{r_\theta r_y + A_y \sqrt{r_y^2 + r_\theta^2 - A_y^2}}{A_y^2 - r_\theta^2} \right) \\ \arctan \left( \frac{-r_x r_z - A_x \sqrt{r_x^2 + r_z^2 - A_x^2}}{A_x^2 - r_z^2} \right) \\ \arctan \left( \frac{v_{GPS}}{u_{GPS}} \right) \end{bmatrix} \quad (4.50)$$

where  $\phi_{m_1}$  was chosen instead of  $\phi_{m_2}$  without loss of generality. The measurements vector calculation is performed every time new GPS data is available. The choice between the plus or minus signals on Eqs. (4.45) and (4.48) can be decided based on simulation results, and it turns out to be signals expressed

on Eq. (4.50). Recall Eq. (3.4b), the Observation matrix of the Kalman filter becomes linear with these analytical solutions:

$$H(k+1) = [ I_{3 \times 3} \mid 0_{3 \times 3} ] \quad (4.51)$$

Moreover, the  $\underline{z}$  calculation has to be performed in a *cascade* way: first the yaw angle  $\psi_m$  is computed, then with this angle computed it is possible to find the pitch angle  $\theta_m$  and finally,  $\phi_m$  can be obtained. However, some disadvantages upon the implementation of this measurement model might arise, such as:

- The errors are accumulated due to the cascaded structure. For instance, the pitch angle  $\theta_m$  is obtained with a version  $\psi_m$  that already has noise.
- The various white noises might no longer keep their *Gaussianity* after undergoing the nonlinear transformations/calculations.

#### 4.2.4 Measurement Model 2

Newton's laws can be used to express the Body-fixed frame accelerations in terms of gravity, angular rates and axial accelerations as presented in Section 4.2.1. If we divide Eq. (4.21) by the mass, the following is obtained:

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_B = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}_B + \begin{bmatrix} p \\ q \\ r \end{bmatrix}_B \times \begin{bmatrix} u \\ v \\ w \end{bmatrix}_B - \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} g \quad (4.52)$$

where all the terms are already expressed in the Body-fixed reference frame  $\mathcal{F}_B$ . This model will additionally presume the availability of airspeed measurements. With the definition of the airspeed vector  $V_a$  in Section 2.1.4, the velocity in the Body-fixed reference frame  $\mathcal{F}_B$  can be obtained with the rotation matrix of Eq. (2.11), resulting in:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix}_B = V_a \begin{bmatrix} \cos \alpha \cos \beta \\ \sin \beta \\ \sin \alpha \cos \beta \end{bmatrix} \quad (4.53)$$

where  $\alpha$  and  $\beta$  were already defined as the angle of attack and sideslip angle, respectively. The  $V_a$  is the True Airspeed obtained through an airspeed sensor. The following is going to be assumed through the derivation of this measurement model:

$$\theta \approx \alpha + \gamma \quad \text{and} \quad \beta \approx 0 \quad (4.54)$$

where  $\gamma$  was already defined as the flight path angle. Hence, the following is obtained:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix}_B = V_a \begin{bmatrix} \cos \alpha \\ 0 \\ \sin \alpha \end{bmatrix} = V_a \begin{bmatrix} \cos \theta \cos \gamma + \sin \theta \sin \gamma \\ 0 \\ \sin \theta \cos \gamma - \cos \theta \sin \gamma \end{bmatrix} \quad (4.55)$$

This idea was previously implemented in UAV/MAV attitude determination, e.g., in (Eldredge, 2006). However, it is very common to make more assumptions such as the climb angle  $\gamma$  and the accelerations  $[\dot{u}, \dot{v}, \dot{w}]^T$  to be zero. Consequently, this substantially reduces the allowed maneuvers to be performed by the UAV. If these assumptions are not adopted and we time-differentiate Eq. (4.55), we obtain:

$$\begin{aligned} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}_B &= V_a \begin{bmatrix} -\dot{\theta} \sin \theta \cos \gamma - \dot{\gamma} \cos \theta \sin \gamma + \dot{\theta} \cos \theta \sin \gamma + \dot{\gamma} \sin \theta \cos \gamma \\ 0 \\ \dot{\theta} \cos \theta \cos \gamma - \dot{\gamma} \sin \theta \sin \gamma + \dot{\theta} \sin \theta \sin \gamma - \dot{\gamma} \cos \theta \cos \gamma \end{bmatrix} \\ &\quad + \dot{V}_a \begin{bmatrix} \cos \theta \cos \gamma + \sin \theta \sin \gamma \\ 0 \\ \sin \theta \cos \gamma - \cos \theta \sin \gamma \end{bmatrix} \end{aligned} \quad (4.56)$$

Developing the second term on the right side of Eq. (4.52) as well:

$$\begin{aligned} \begin{bmatrix} p \\ q \\ r \end{bmatrix}_B \times \begin{bmatrix} u \\ v \\ w \end{bmatrix}_B &= \begin{bmatrix} qw - rv \\ ru - pw \\ pv - qu \end{bmatrix} = \begin{bmatrix} qw \\ ru - pw \\ -qu \end{bmatrix} \\ &= V_a \begin{bmatrix} q(\sin \theta \cos \gamma - \cos \theta \sin \gamma) \\ r(\cos \theta \cos \gamma + \sin \theta \sin \gamma) - p(\sin \theta \cos \gamma - \cos \theta \sin \gamma) \\ -q(\cos \theta \cos \gamma + \sin \theta \sin \gamma) \end{bmatrix} \end{aligned} \quad (4.57)$$

Eq. (4.52) finally becomes:

$$\begin{aligned} \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} &= V_a \begin{bmatrix} -\dot{\theta} \sin \theta \cos \gamma - \dot{\gamma} \cos \theta \sin \gamma + \dot{\theta} \cos \theta \sin \gamma + \dot{\gamma} \sin \theta \cos \gamma \\ 0 \\ \dot{\theta} \cos \theta \cos \gamma - \dot{\gamma} \sin \theta \sin \gamma + \dot{\theta} \sin \theta \sin \gamma - \dot{\gamma} \cos \theta \cos \gamma \end{bmatrix} + \dot{V}_a \begin{bmatrix} \cos \theta \cos \gamma + \sin \theta \sin \gamma \\ 0 \\ \sin \theta \cos \gamma - \cos \theta \sin \gamma \end{bmatrix} \\ &\quad + V_a \begin{bmatrix} q(\sin \theta \cos \gamma - \cos \theta \sin \gamma) \\ r(\cos \theta \cos \gamma + \sin \theta \sin \gamma) - p(\sin \theta \cos \gamma - \cos \theta \sin \gamma) \\ -q(\cos \theta \cos \gamma + \sin \theta \sin \gamma) \end{bmatrix} - \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} g \end{aligned} \quad (4.58)$$

Taking the first equation of the previous expression and solving it for  $\theta$ :

$$\begin{aligned} A_x &= V_a(-\dot{\theta} \sin \theta \cos \gamma - \dot{\gamma} \cos \theta \sin \gamma + \dot{\theta} \cos \theta \sin \gamma + \dot{\gamma} \sin \theta \cos \gamma) \\ &\quad + \dot{V}_a(\cos \theta \cos \gamma + \sin \theta \sin \gamma) + V_a q(\sin \theta \cos \gamma - \cos \theta \sin \gamma) + g \sin \theta \\ \Leftrightarrow A_x &= \underbrace{(-V_a \dot{\theta} \cos \gamma + V_a \dot{\gamma} \cos \gamma + V_a q \cos \gamma + \dot{V}_a \sin \gamma + g)}_{a_1} \sin \theta \\ &\quad + \underbrace{(-V_a \dot{\gamma} \sin \gamma + V_a \dot{\theta} \sin \gamma - V_a q \sin \gamma + \dot{V}_a \cos \gamma)}_{a_2} \cos \theta \\ \Leftrightarrow A_x &= a_1 \sin \theta + a_2 \cos \theta \end{aligned} \quad (4.59)$$

that corresponds to the same equation structure that was analytically solved in the last section (recall Eq. (4.45) and Eq. (4.48)). Therefore, solving the same manner, the solution for pitch measured angle  $\theta_m$  can be obtained:

$$\theta_m = \arctan \left( \frac{a_1 a_2 \pm A_x \sqrt{a_1^2 + a_2^2 - A_x^2}}{A_x^2 - a_1^2} \right) \quad (4.60)$$

with:

$$\begin{aligned} a_1 &= -V_a \dot{\theta} \cos \gamma + V_a \dot{\gamma} \cos \gamma + V_a q \cos \gamma + \dot{V}_a \sin \gamma + g \\ a_2 &= -V_a \dot{\gamma} \sin \gamma + V_a \dot{\theta} \sin \gamma - V_a q \sin \gamma + \dot{V}_a \cos \gamma \end{aligned} \quad (4.61)$$

The solution for the roll measured angle  $\phi_m$  can be derived as well. From the second expression of Eq. (4.58) the solution is given as:

$$\phi_m = \arcsin \left( -\frac{A_y - V_a r(\cos \theta \cos \gamma + \sin \theta \sin \gamma) + V_a p(\sin \theta \cos \gamma - \cos \theta \sin \gamma)}{g \cos \theta} \right) \quad (4.62)$$

The solution for the pitch and roll measured angles was found. The measured yaw/course angle  $\psi_m$  can be obtained with the GPS velocity, as in Section 4.2.3, Eq. (4.36), taking advantage of its relatively high precision.

As happened with the Measurement Model 1 deduced in the last section, the current measurement model also runs on GPS availability. The required parameters in Eqs. (4.60) and (4.62) can be obtained as follows:

- $V_a$  is obtained with an airspeed sensor as previously said. The airspeed sensor output has a higher frequency than the GPS unit. However, it must be used with GPS data; then it can be differentiated over longer periods of time, reducing the high frequency noise differentiation disadvantage. The  $\dot{V}_a$  is provided as follows:

$$\dot{V}_a(k_{GPS}) = \frac{V_a(k_{GPS}) - V_a(k_{GPS}-1)}{\Delta t_{GPS}} \quad (4.63)$$

where  $k_{GPS}$  and  $k_{GPS}-1$  are the new and previous GPS measurements instants, respectively. The time difference between the two GPS measurements instants is given by  $\Delta t_{GPS}$ .

- The flight path angle  $\gamma$  can be obtained by:

$$\gamma = \arctan\left(\frac{-w_{GPS}}{V_{GPS_{Horiz}}}\right) = \arctan\left(\frac{-w_{GPS}}{\sqrt{u_{GPS}^2 + v_{GPS}^2}}\right) \quad (4.64)$$

where  $V_{GPS_{Horiz}}$  is the velocity in the local horizontal plane. Another way to get the flight path angle is through the use of a barometric pressure sensor, but this alternative is not going to be further discussed<sup>5</sup>.

- The flight path angle *rate*  $\dot{\gamma}$  can be obtained by time-differencing Eq. (4.64):

$$\dot{\gamma} = \frac{-V_{GPS_{Horiz}}\dot{w}_{GPS} + w_{GPS}\dot{V}_{GPS_{Horiz}}}{w_{GPS}^2 + V_{GPS_{Horiz}}^2} \quad (4.65)$$

or simply by finite differences:

$$\dot{\gamma} = \frac{\gamma(k_{GPS}) - \gamma(k_{GPS}-1)}{\Delta t_{GPS}} \quad (4.66)$$

The latter is the chosen method due to simplicity and to avoid calculation of GPS accelerations.

- The time-derivative of pitch Euler angle,  $\dot{\theta}$ , is present in the EKF but not in the *correction* step. However, its addition in this step could be beneficial, as can be seen from its presence in Eq. (4.60), and it can be brought from the *prediction* step.

This measurement model has the Euler angles as measurements as well. The vector  $\underline{z}$  is given by:

$$\underline{z} = \begin{bmatrix} \phi_m \\ \theta_m \\ \psi_m \end{bmatrix} = \begin{bmatrix} \arcsin\left(-\frac{A_y - V_a r(\cos \theta \cos \gamma + \sin \theta \sin \gamma) + V_a p(\sin \theta \cos \gamma - \cos \theta \sin \gamma)}{g \cos \theta}\right) \\ \arctan\left(\frac{a_1 a_2 - A_x \sqrt{a_1^2 + a_2^2 - A_x^2}}{A_x^2 - a_1^2}\right) \\ \arctan\left(\frac{v_{GPS}}{u_{GPS}}\right) \end{bmatrix} \quad (4.67)$$

where  $a_1$  and  $a_2$  were previously defined in Eq. (4.61). This measurement model also works in a *cascade* form: first the yaw angle  $\psi_m$  is computed, then with this obtained angle it is possible to find the pitch angle  $\theta_m$  and finally, the  $\phi_m$  can be calculated. However, as the other model some disadvantages may arise upon its implementation, such as the ones from the Measurement Model 1. Additionally, this model might be more noise sensitive due to a larger number of differentiations. Finally, the true *nature* of angle of attack is not captured due to the assumption of Eq. (4.54). For instance, if the UAV is banked, it could be experiencing a high angle of attack with a zero pitch and zero climb angle. The Observation matrix of the Kalman filter is linear and given by Eq. (4.51) as well.

---

<sup>5</sup>An example of integration of a barometric pressure sensor in the measurements model of an EKF for UAV attitude determination with GPS/INS data fusion can be found in (Eldredge, 2006).

#### 4.2.5 Other implementation issues

Since one of the objectives is to estimate the attitude with the lowest computation effort possible, some simplifications can be useful upon the KF implementation. The EKF's *one step ahead prediction*, Eq. (3.30), can be carried using the *Euler integration method*:

$$\hat{x}(k+1|k) = \hat{x}(k|k) + \underline{f}(\underline{x}(t_k), \underline{u}^*(t_k))\Delta t \quad (4.68)$$

where  $\Delta t$  is the sampling time. This is the most simple method for numerical integration of ordinary differential equations. More efficient and complex methods could be used, such as Runge-Kutta fourth-order (Ascher & Petzold, 1998). However, the Euler integration method of Eq. (4.68) will most probably be enough.

Continuing the line of thought about computation effort reduction, the discretization step of the Kalman filter, Eq. (3.19) is rather power demanding if many orders are obtained. Thus, it will be considered that only the first three terms are enough to properly calculate the System transition matrix:

$$\Phi(k) = e^{F(t_k)\Delta t} \approx I + F(t_k)\Delta t + \frac{(F(t_k)\Delta t)^2}{2!} \quad (4.69)$$



# Chapter 5

## Tests and Results

This chapter will present the results of the derived algorithms for simulation and real flights. Firstly, the simulation environment is presented in Section 5.1 where other additional issues are also discussed. Then, in Section 5.2 the algorithms are tested with two simulation flights. Finally, one of the algorithms is tested with real flight data in Section 5.3.

### 5.1 Simulation Environment

This section discusses the simulation *environment*, i.e., the characteristics of the sensors, the time-synchronization between sensors and the simulation profiles chosen to test the algorithms in the next chapter.

Simulation is a very important feature when developing new software. When running a test on UAV's hardware, there is no suitable measure of the true state. It is common to compare results to a larger, higher power reference system, which is almost impossible on this type of small platforms. Nevertheless, it still does not provide a true measure of the states. Simulation has the advantage that the true state values are available for comparison. The emulated sensor measurements can be logged with the estimated and true states with the additional variables that might be useful to detect model deficiencies.

This sections begins with a brief description about the simulation program used to obtain the simulated UAV's flight data, in Section 5.1.1. Then, the GPS/INS synchronization issue, already referred upon the integration strategies, is going to be addressed in Section 5.1.2. The simulation environment will be described in detail with the sensors characteristics in Section 5.1.3 and the simulated flights in Section 5.1.4.

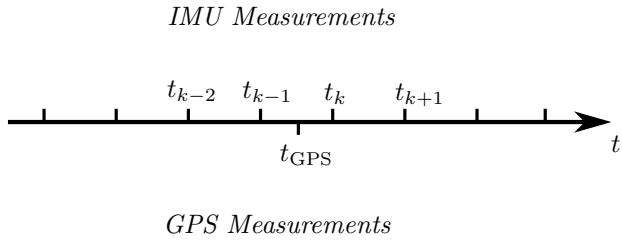
Finally, the Section 5.2 will contain the results from the simulations performed comparing its performance to the true state data from the simulator.

#### 5.1.1 Aeronautical Simulation Library

The fixed-wing UAV model that is used in this thesis is based on the Aeronautical Simulation MATLAB/Simulink library from (Unmanned Dynamics LLC, 2002). The AeroSim aeronautical simulation blockset provides a set of tools for the development of nonlinear 6-degree-of-freedom aircraft dynamic models. In addition to the basic aircraft dynamics blocks, the library also includes complete aircraft models which can be customized through parameter files. The practical implementation examples provided with the AeroSim blockset include the *Navion* - a general-aviation airplane, and the *Aerosonde* UAV - a long-range weather-reconnaissance autonomous airplane, that will be used as simulation model.

### 5.1.2 GPS/INS Synchronization

Time synchronization between GPS and INS measurements becomes a problem when using an integrated GPS/INS sensor fusion algorithm. When multiple sensors are used, each producing its own data, synchronization among all the systems is required. The GPS receiver and inertial sensors are separate, self-contained subsystems with different frequencies and processing times (see Figure 5.1). Additionally, when dealing with fast dynamics UAVs, the time synchronization requirement becomes even more important as the innovation may take place too late with respect to the prediction step (Hummelink, 2011). The time synchronization issue can be improved by adding an additional state to the integration filter, as can be found in (Skog & Handel, 2010). Additional time synchronization methods exist and some of them can be found e.g., (Ding, Wang, Li, Mumford, & Rizos, 2008).



**Figure 5.1:** IMU and GPS output frequencies are different and require synchronization.

Some GPS receivers provide the 1 Pulse-Per-Second (PPS) signal. This electrical signal indicates the turnover time of each second and can be used as a timing reference to synchronize the GPS signals with the IMU. Considering the short term stability and accuracy of the clock oscillator acceptable, the relationship between the GPS and INS clocks can be traced by checking the time difference between the GPS 1PPS time and the INS pulse time.

In general, the sampling instants are not coordinated between GPS and INS, therefore guaranteed accuracy depends on the INS sampling rate. Considering that the INS has a sampling rate of 100Hz (the same as the device used to obtain the real data in Section 5.3), the guaranteed synchronization accuracy is 0.005 seconds. Sometimes an interpolation method can be used to improve the virtual sampling rate of the INS (Shin, 2001). If a highly dynamic vehicle with low INS sampling rate is utilized, this is strongly recommended.

In the simulation profiles the time synchronization is not a concern. However, for the real data implementation, the 1PPS GPS's signal was utilized to synchronize both navigation systems.

### 5.1.3 Sensors Characteristics

In order to properly simulate the developed algorithms, noise and bias should be introduced in the virtual sensors. This way, the simulation will be more *similar* to a real environment. The sensors were previously modeled in Chapter 4 and their noise and bias values are now presented in Table 5.1. Some additional information about the sensors should be also presented:

- The values from Table 5.1 are close to the specified performance in the manual of real sensors as the *MTi-G Xsens*<sup>1</sup> device that will be used to obtain real flight data in Section 5.3.
- In order to model the bias-drift as a first-order Gauss-Markov process, information about real values for this parameter was taken from the *Crista IMU* because the MTi-G Xsens devices does

<sup>1</sup>For more information about the MTi-G Xsens please refer to Appendix E.1 or its manual, (Xsens Technologies, 2009)

not provide a complete documentation. Thus, the Gauss-Markov gyroscopes parameters have the following values (Solimeno, 2007; Godha, 2006):  $\sigma := \sigma_{b_w} = 211$  deg/h and  $\tau_c = 382$  s.

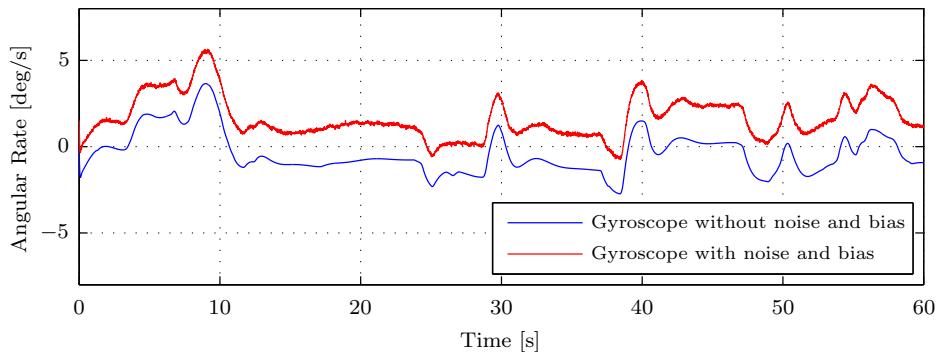
- As previously said, the biases of the accelerometers are generally small and they were assumed to be zero in the simulations without loss of generality.
- In addition to the bias-drift and white Gaussian noise introduced in the measurements of the gyroscopes, a *turn-on* bias was also included. This value was taken arbitrarily. Figure 5.2 shows an example of how a signal from a gyroscope becomes after all the noise and biases introduced.
- The airspeed sensor was not individually modeled but, when properly calibrated, it can be seen as a bias-free sensor where the airspeed real values  $V_a$  are affected by white Gaussian noise  $\varepsilon_{V_a}$ :

$$V_{a_m} = V_a + \varepsilon_{V_a} \quad (5.1)$$

where  $V_{a_m}$  is the measured airspeed.

**Table 5.1:** Noise and bias introduced in the simulation sensors.

	Symbol	Value
Accelerometer noise	$\sigma_a$	0.02 m/s <sup>2</sup>
Gyroscope noise	$\sigma_w$	0.05 deg/s
Gyroscope <i>turn-on</i> bias	$b_0$	1.5 deg/s
Gyroscope bias stability	$\sigma_{b_w}$	211 deg/h
GPS velocity noise	$\sigma_{V_{GPS}}$	0.01 m/s
Airspeed noise	$\sigma_{V_a}$	0.06 m/s



**Figure 5.2:** Example of a gyroscope signal after noise and bias introduction from Table 5.1.

#### 5.1.4 Simulation Flights

The simulation flights are now presented where **two different flight simulations were conducted to test the two different EKF models**. Recall that both EKF derived algorithms have as Process model the structure derived in Section 4.2.2 and they differ in the measurements model. The EKF<sub>1</sub> has the measurement model described in Section 4.2.3 and the EKF<sub>2</sub> the measurement algorithm derived in Section 4.2.4.

The sideslip angle and angle of attack can not be measured with the sensors used in the UAV. However, they are presented in Appendix C. They will be useful to understand the flaws of the implementation, i.e., to detect when and why the algorithm loses estimation capability.

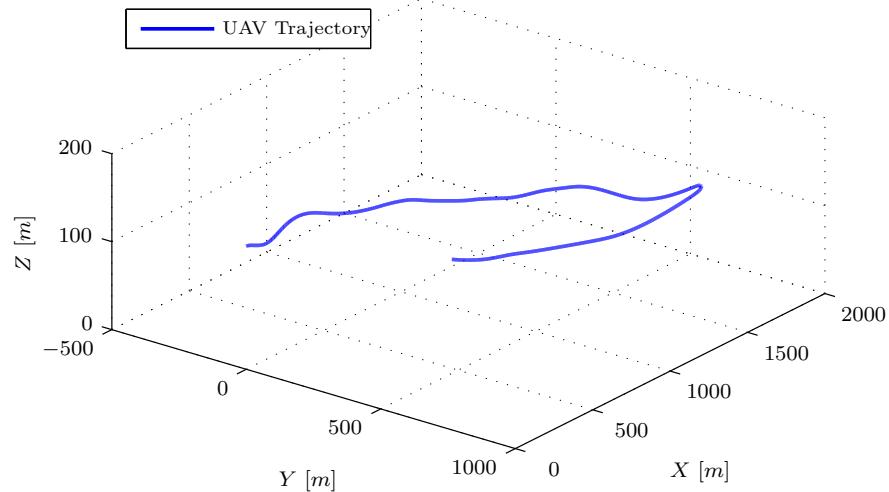
The simulated flights have the following characteristics:

- **Flight 1:**

- Duration: 100 s.
- All kind of maneuvers were performed representing a general flight. The airplane turns right pitching up for a while, then turns left and starts descending varying its airspeed (see Figures C.2 and C.3). The 3D trajectory of the UAV is presented in Figure 5.3.
- The sideslip angle  $\beta$  was kept low.
- All the flight data is presented in Appendix C for a complete flight description.

- **Flight 2:**

- Duration: 100 s.
- This flight was mainly performed in order to test the limits of the derived EKF algorithms. The airplane performs a 360 degrees turn, but its altitude and airspeed is almost constant (see Figures C.5 and C.6). Highly asymmetrical flight maneuvers are performed. The 3D trajectory of the UAV is presented in Figure 5.4.
- The sideslip angle  $\beta$  reaches higher values.
- All the flight data is presented in Appendix C for a complete flight description.



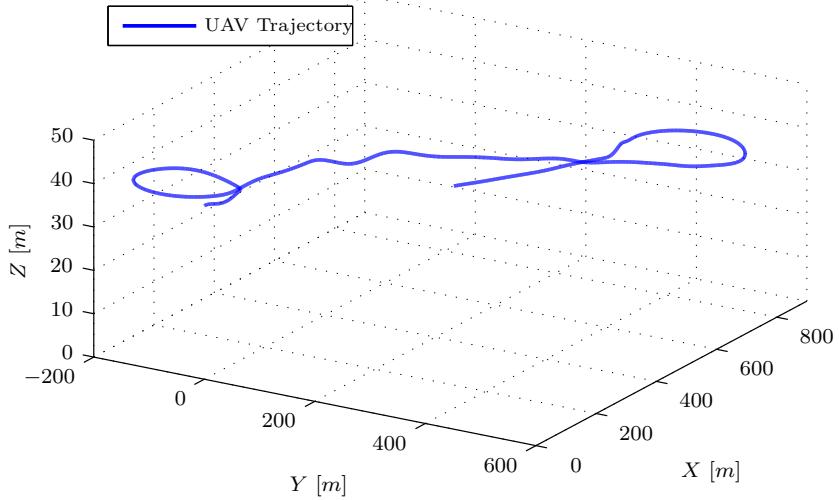
**Figure 5.3:** 3D position of Flight 1. Initial position: [0, 0, 140]m.

### Kalman Filter Tuning

The initial parameters of the EKF have to be defined in order to perform a correct estimation. Firstly, the initial state estimation  $\hat{x}$  is chosen as all states equal to zero:

$$\hat{x}(0|0) = [0, 0, 0, 0, 0, 0]^T \quad (5.2)$$

The initial condition for the error covariance matrix  $P(0|0)$  ( $6 \times 6$  matrix) can be large for unknown initial state error or small for a large confidence in the initial state estimation. Considering moderate



**Figure 5.4:** 3D position of Flight 2. Initial position: [0, 0, 40]m.

confidence in the initial state estimation (errors of 5 deg in the Euler angles and 2 deg/s in the bias of the gyroscopes) and that no correlated errors exist, the  $P(0|0)$  matrix becomes:

$$P(0|0) = \text{diag} \left( \left[ \left( \frac{5\pi}{180} \right)^2, \left( \frac{5\pi}{180} \right)^2, \left( \frac{5\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2 \right] \right) \quad (5.3)$$

The process noise covariance matrix  $Q$  ( $6 \times 6$  matrix) can be defined based on the noise parameters assigned to the sensors on Table 5.1. Furthermore, it is also considered a diagonal matrix:

$$\begin{aligned} Q &= \text{diag} \left( \left[ \sigma_w^2, \sigma_w^2, \sigma_w^2, \frac{2\sigma_{bw}^2}{\tau_c}, \frac{2\sigma_{bw}^2}{\tau_c}, \frac{2\sigma_{bw}^2}{\tau_c} \right] \right) \\ &= \text{diag} \left( \left[ \left( \frac{0.05\pi}{180} \right)^2, \left( \frac{0.05\pi}{180} \right)^2, \left( \frac{0.05\pi}{180} \right)^2, \frac{2 \left( \frac{211\pi}{3600 \times 180} \right)^2}{382}, \frac{2 \left( \frac{211\pi}{3600 \times 180} \right)^2}{382}, \frac{2 \left( \frac{211\pi}{3600 \times 180} \right)^2}{382} \right] \right) \end{aligned} \quad (5.4)$$

The measurements noise covariance matrix  $R$  ( $3 \times 3$  matrix) has not a straightforward formulation as the process noise covariance matrix  $Q$ . This is due to the analytical solutions obtained for the measurements models of both EKF<sub>1</sub> and EKF<sub>2</sub>. As it was said in Section 4.2, the various white noises might no longer keep their Gaussianity after undergoing the nonlinear transformations/calculations, and the measurements Eq. (3.29) might not hold. A more *trial and error* approach to define the  $R$  matrix is required, adjusting its diagonal entries manually with the help of simulations. For instance, considering a standard deviation of  $\sigma_\phi = \sigma_\theta = \sigma_\psi = 1$  deg for all the Euler angles:

$$\begin{aligned} R &= \text{diag} \left( [\sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2] \right) \\ &= \text{diag} \left( \left[ \left( \frac{1\pi}{180} \right)^2, \left( \frac{1\pi}{180} \right)^2, \left( \frac{1\pi}{180} \right)^2 \right] \right) \end{aligned} \quad (5.5)$$

However, an investigation to verify if a Gaussian distribution still fits the measurements error/noise was performed and is presented in Appendix B. Results show that the distribution remains relatively Gaussian even under the nonlinear transformations/calculations of the measurements models of Sections 4.2.3 and 4.2.4. These results also provided a fair initial starting point to achieve the final  $R$  matrix, presented in Eq. (5.5).

Finally, regarding the characteristics of the bias of the gyroscopes it might be desirable to increase their estimation's convergence speed (Gebre-Egziabher et al., 2004). Recall Eq. (4.32), from the EKF process modeling, that can also be written as:

$$G[\underline{x}(t), t] = \begin{bmatrix} F_{x_{(1:3,4:6)}} & 0_{3 \times 3} \\ 0_{3 \times 3} & \lambda I_{3 \times 3} \end{bmatrix} \quad (5.6)$$

where the parameter  $\lambda$  was introduced and  $F_{x_{(1:3,4:6)}}$  corresponds to a part of Eq. (4.34). It is possible to change the bias estimator poles by changing  $\lambda$ . If the parameter  $\lambda$  is smaller than unity, the convergence speed is reduced. On the other hand, a bigger value for  $\lambda$  ( $> 1$ ) results in faster poles of the bias estimator of the gyroscopes, which leads to an increased speed of convergence. However, this also results in a noisier solution. Thus, there is a trade-off involved when selecting values for this tuning parameter so that the bias estimation convergence speed increases without significantly magnifying the noise on the Euler angles estimate. Since UAVs are considered highly dynamic aircraft it was chosen to increase the speed of convergence by one order of magnitude:

$$\lambda = 10 \quad (5.7)$$

### 5.1.5 Other Simulation Issues

For both simulation and real flights there are still two issues that should be referred:

- *Heading observability:* In situations with relatively low dynamics, the heading might not be observed from the GPS velocity measurements (remember Eq. (4.36)). This situation could happen for instance when the UAV is flying with strong (front) wind, where the GPS velocity (*ground velocity*) might get close to zero. To improve the heading observability several strategies can be used: additional aiding sensors (magnetometers to determine the yaw of the aircraft or air data sensors to estimate the wind) or making assumptions about the probability of the motion of the UAV for a certain situations (the GPS update could also be stopped upon this situation as if it were a GPS *non-fix*). Nevertheless, the heading non-observability is an utmost situation, not that common in fixed-wing aircraft.
- *Pitch angle singularity:* As we can see from Eq. (4.31), the Process model presents a singularity for  $\theta = 90$  deg. This situation is commonly referred as (Euler angles) *gimbal lock* and it adds a limitation to the allowed maneuvers that the UAV can perform:

$$-90 \text{ deg} < \theta < 90 \text{ deg} \quad (5.8)$$

This singularity could be solved with the use of quaternions and it is specially recommend for space-craft attitude parameterization. It is important to be aware of this situation, however, reaching pitch angles of 90 deg is not that frequent, even for small UAVs, and the Euler attitude representation seems adequate enough.

## 5.2 Simulation Results

In this chapter the EKF<sub>1</sub> and EKF<sub>2</sub> simulation results are presented. Recall that EKF<sub>1</sub> and EKF<sub>2</sub> have the same dynamics model and they differ in the measurements model. The EKF<sub>1</sub> focus mainly on the GPS-derived acceleration and the EKF<sub>2</sub> makes use of an airspeed sensor. The following additional assumptions were made:

- The gravity magnitude is considered constant:  $g = 9.80665 \text{ m/s}^2$ .
- The output data frequency of the gyroscopes and accelerometers is equal to 100 Hz.
- The output data frequency of the GPS velocity is equal to 10 Hz.
- There is no wind.

In Section 5.2.1 the simulation results of the EKF<sub>1</sub> for the two flights is presented. The analysis of the EKF<sub>2</sub> is analogous in Section 5.2.2. Then, in Section 5.2.3, GPS outages are introduced to observe the estimation performance when only the INS is operating. Finally, in Section 5.2.4 a summary about the results is presented and the advantages and flaws of the two models are discussed.

### 5.2.1 Results of EKF<sub>1</sub>

The EKF<sub>1</sub> performance is going to be evaluated with the two flights already described in the last chapter and the parameters defined in Section 5.1.4. Recall that the additional information about the two flights is presented in Appendix C. A reliability testing of the EKF<sub>1</sub> for both flights is also provided.

#### Flight 1

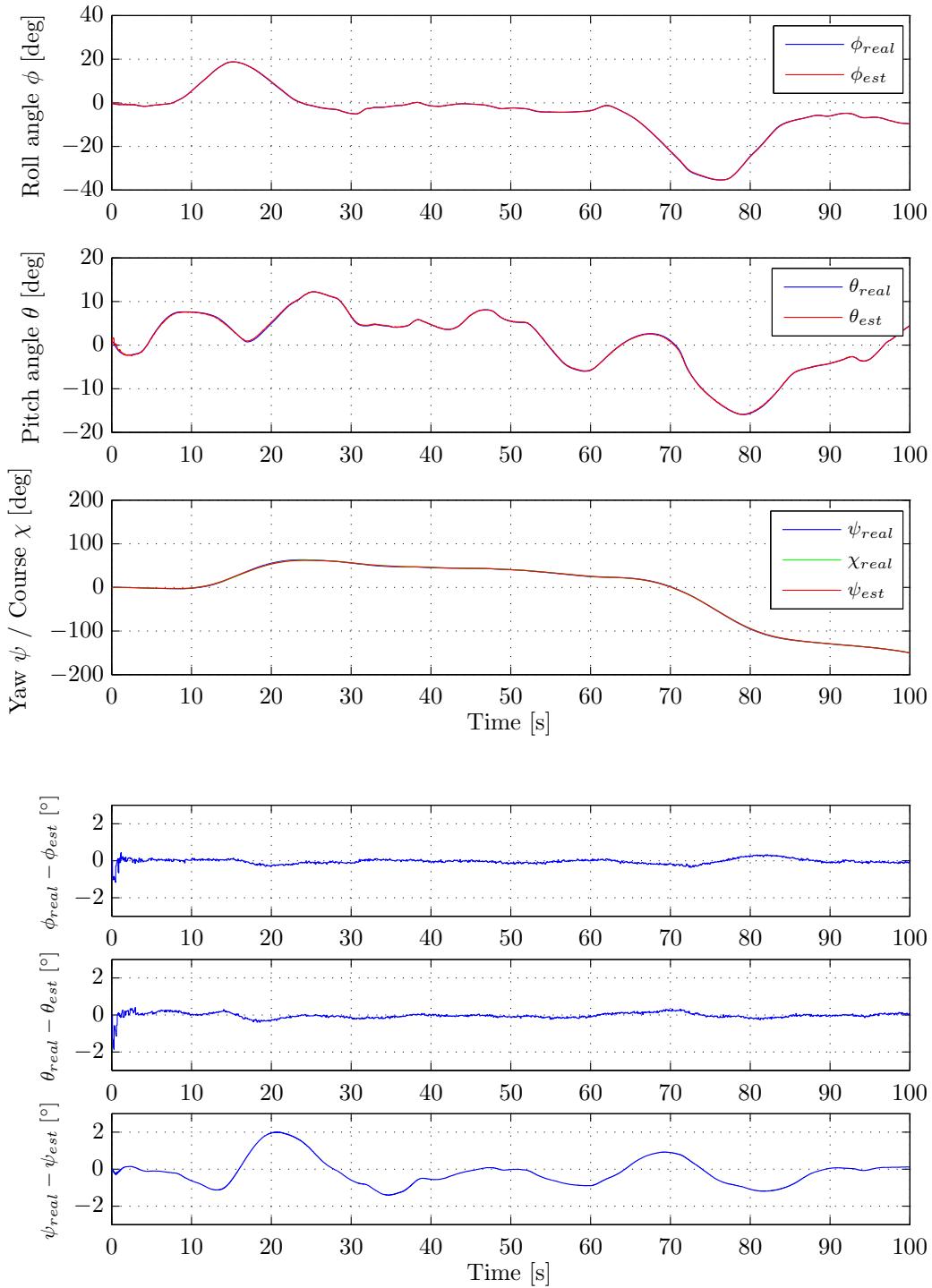
Figure 5.5 shows the estimated attitude angles roll  $\phi$ , pitch  $\theta$  and yaw  $\psi$  of the UAV. As it is possible to see, for all cases the estimation error is very small. The attained attitude angles are varied where the roll angle reach magnitudes of 36 degrees and the pitch angle values of 16 degrees. The difference between the true and estimated attitude angles is also presented in Figure 5.5.

We can notice the fast initial convergence for the three Euler angles. After that convergence, the EKF<sub>1</sub> algorithm keeps the estimation error very small during all simulation time. The biggest estimation errors occur for the yaw angle  $\psi$ . This can be explained due to the assumption that the sideslip angle is zero. As we can see from Figure C.2, this assumption is not always true and the yaw estimation error is congruent with this: when the sideslip angle increases, the yaw estimation error is higher. For a matter of curiosity, the course angle  $\chi$  is also presented in Figure 5.5.

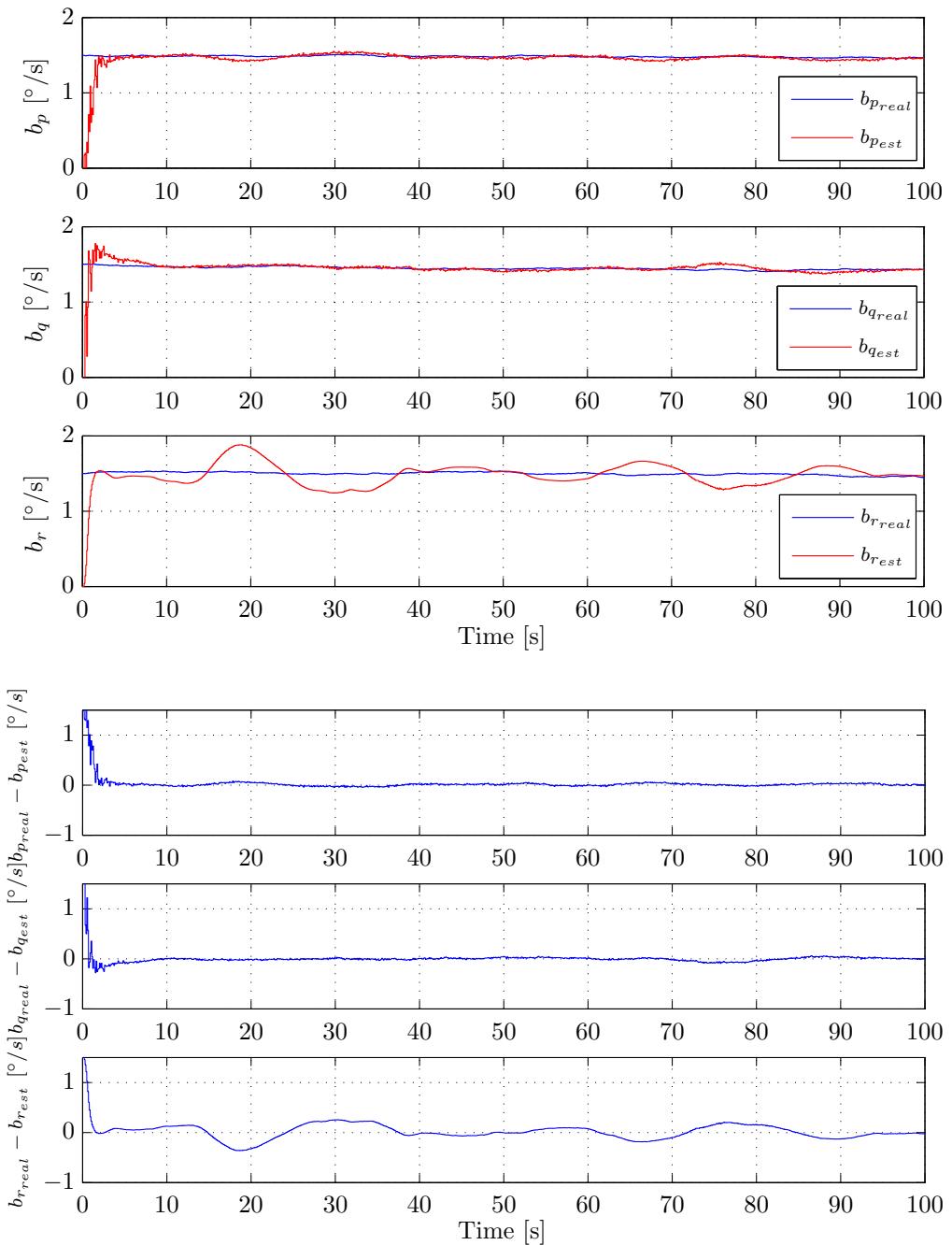
Figure 5.6 shows the estimated bias for the three gyroscopes. The initial convergence to the *turn-on* bias of 1.5 degrees is noticeable. After that convergence, the bias estimation remains close to that value. The only bias that shows a bigger estimation error is  $b_r$ . We can see the relation between a bigger estimation error and the existence of a sideslip angle. The difference between the real and estimated bias is presented in Figure 5.6 as well.

Table 5.2 presents the mean error and standard deviation between the true and the estimated states. The mean error is close to zero for all the estimated states. Additionally, all the standard deviations are very small (smaller than 1 degree). This means that the states estimation is very close to the real values of the simulation.

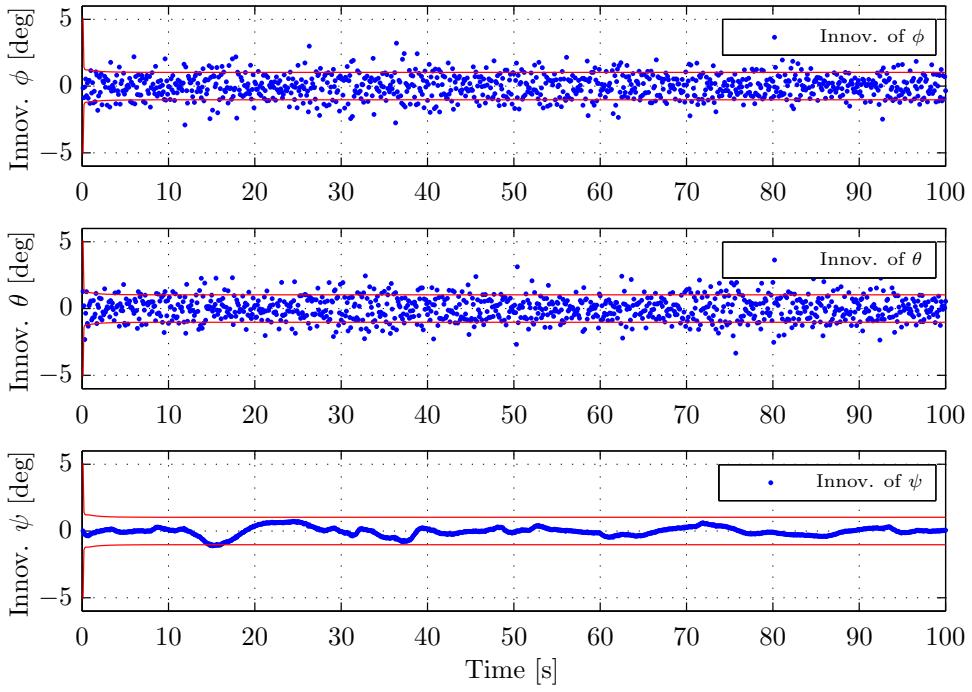
In order to test the reliability of the KF, the analysis of the innovation sequence can be performed (as mentioned in Section 3.2.7). The innovation sequences of the EKF<sub>1</sub> are presented in Figure 5.7 and the corresponding mean and standard deviation are presented in Table 5.3. As we can see, the innovation sequences of  $\phi$  and  $\theta$  resemble white noise and their values are mainly within the standard deviation of the innovation. On the other hand, the innovation sequence of the  $\psi$  is somewhat correlated. Nevertheless, its mean is very close to zero and the standard deviation is the lowest one.



**Figure 5.5:** Estimation of  $\phi$ ,  $\theta$  and  $\psi$ , and differences between real and estimated angles, with EKF<sub>1</sub>, Flight 1.



**Figure 5.6:** Estimation of  $b_p$ ,  $b_q$  and  $b_r$ , and differences between real and estimated bias with EKF<sub>1</sub>, Flight 1.



**Figure 5.7:** Innovation of the three Euler angles with EKF<sub>1</sub> for Flight 1. The red line represents the standard deviation of the Innovation.

**Table 5.2:** Numerical results of EKF<sub>1</sub> for Flight 1.

State Error	Mean	Std. Deviation
$\phi$	-0.039 deg	0.135 deg
$\theta$	-0.024 deg	0.136 deg
$\psi$	-0.150 deg	0.759 deg
$b_p$	0.027 deg/s	0.124 deg/s
$b_q$	-0.001 deg/s	0.094 deg/s
$b_r$	0.022 deg/s	0.165 deg/s

**Table 5.3:** Innovation results of EKF<sub>1</sub> for Flight 1.

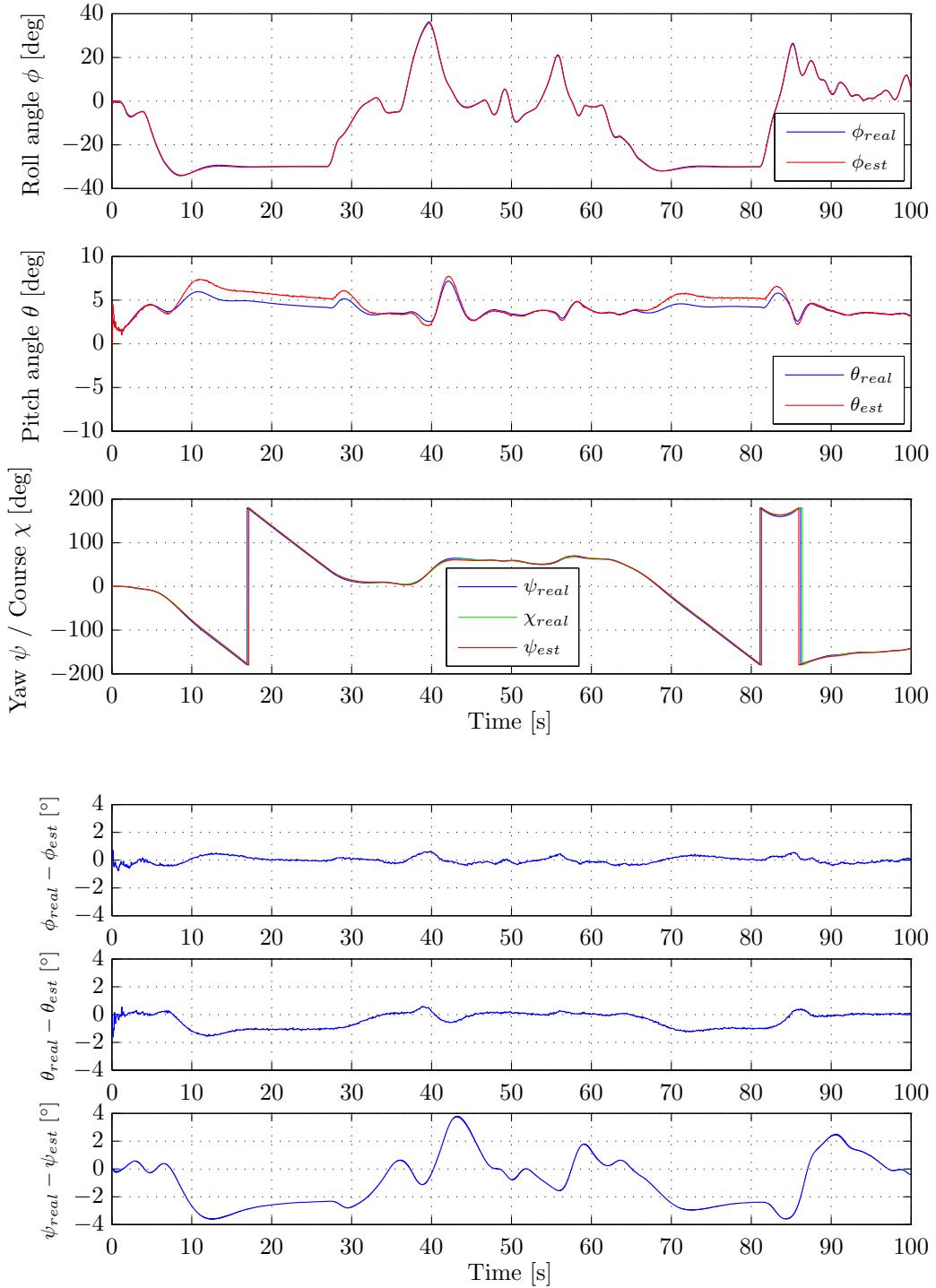
Innov. Sequence	Mean [deg]	Std. Deviation [deg]
$\phi$	-0.029	0.874
$\theta$	-0.015	0.880
$\psi$	-0.030	0.327

## Flight 2

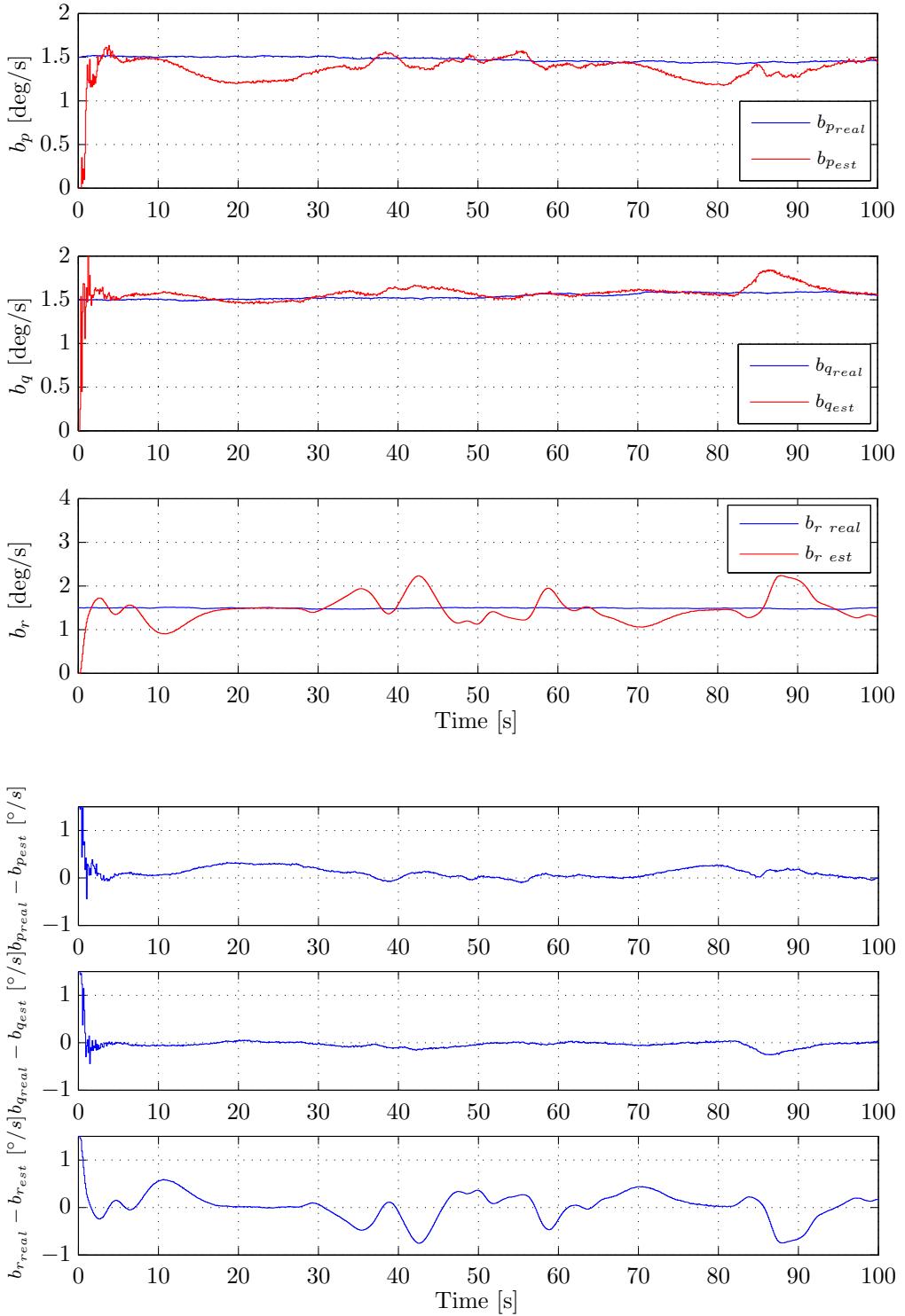
The estimated attitude angles of the UAV and the differences between the truth and estimation are presented in Figure 5.8. The same analogy is taken for the bias estimation in Figure 5.9.

This flight is more demanding in terms of roll angle, where it remains constant at -30 degrees for long periods (16~27s and 72~81s). Additionally, in these periods, the UAV keeps the altitude and airspeed constant and the sideslip angle very close to zero (Figures C.5 and C.6). The UAV is thus performing a coordinated turn. From Figure 5.8, we can see that the pitch angle estimation deteriorates when this type of maneuver is held for long periods. However, this *error* in the pitch estimation remains constant during this activity (around one degree). In Figure D.1, the pitch angle estimation is presented with the Eq. (2.13) verification. A clear relationship exists: when Eq. (2.13) starts losing validity, the pitch angle estimation deteriorates as well.

As for the roll angle estimation, the performance is similar to the one obtained with the first flight, remaining a very accurate estimation. On the other hand, since the sideslip angle reaches bigger values, the yaw angle estimation error is also higher in these situations, which leads to a less accurate estimation.



**Figure 5.8:** Estimation of  $\phi$ ,  $\theta$  and  $\psi$ , and differences between real and estimated angles, with EKF<sub>1</sub>, Flight 2.



**Figure 5.9:** Estimation of  $b_p$ ,  $b_q$  and  $b_r$ , and differences between real and estimated bias with EKF<sub>1</sub>, Flight 2.

The estimation of the bias of the gyroscopes tracks the *turn-on* bias of 1.5 degrees after few seconds. Then, these estimations remain close this value but with bigger fluctuations (specially for  $b_p$ ) upon demanding UAV's maneuvers such as the coordinated turns with high roll angle values. These deviations from the true bias values reach maximum values of 0.3 deg/s as can be seen in Figure 5.9. The error of the  $b_r$  estimation is intrinsically related to the sideslip present in this second flight, Figure C.5.

A particular situation should be noticed. When we observe the yaw estimation error during the periods 16~27s and 72~81s for instance, it is not close zero, even with zero sideslip. This seems to be related to the high roll angle that the UAV is experiencing ( $\sim 30$  degrees). After rolling an angle  $\phi$  about the X-axis of the aircraft, the angle of attack will have projections on both the new Y and Z-axis (Etkin & Reid, 1996). Thus, a *sideslip*  $\beta_{roll}$  appears and can be determined as follows (with the assumption of Eq. (2.13)):

$$\beta_{roll} = \arcsin(\sin \alpha \sin \phi) = \arcsin(\sin(\theta - \gamma) \sin \phi) \quad (5.9)$$

The results of the EKF<sub>1</sub> with the inclusion of Eq. (5.9) in the algorithm are presented in Figures D.2 and D.3. The mean error and standard deviation between the true and the estimated states for the EKF<sub>1</sub> with and without Eq. (5.9) in consideration can be found in Table 5.4.

**Table 5.4:** Numerical results of EKF<sub>1</sub> for Flight 2 with and without inclusion of Eq. (5.9) in the algorithm.

EKF <sub>1</sub> without Eq. (5.9) inclusion			EKF <sub>1</sub> with Eq. (5.9) inclusion		
State Error	Mean	Std. Deviation	State Error	Mean	Std. Deviation
$\phi$	-0.002 deg	0.236 deg	$\phi$	-0.021 deg	0.279 deg
$\theta$	-0.400 deg	0.551 deg	$\theta$	0.169 deg	0.290 deg
$\psi$	-0.900 deg	1.793 deg	$\psi$	-0.157 deg	1.227 deg
$b_p$	0.112 deg/s	0.158 deg/s	$b_p$	0.012 deg/s	0.153 deg/s
$b_q$	-0.035 deg/s	0.099 deg/s	$b_q$	-0.031 deg/s	0.102 deg/s
$b_r$	0.030 deg/s	0.305 deg/s	$b_r$	0.015 deg/s	0.270 deg/s
Attitude angles estimation: Figure 5.8			Attitude angles estimation: Figure D.2		
Bias estimation: Figure 5.9			Bias estimation: Figure D.3		

As we can see, there is a relevant improvement in the pitch and yaw angles estimation for both the mean and standard deviation of the estimation error. As for the other states, they remain almost unchanged with the inclusion of Eq. (5.9) in the estimation algorithm.

The innovation sequence with the EKF<sub>1</sub> for this second flight<sup>2</sup> resembles the one presented in the last section and it can be consulted in Figure D.4. The numerical results are presented in Table 5.5.

**Table 5.5:** Innovation results of EKF<sub>1</sub> for Flight 2.

Innov. Sequence	Mean [deg]	Std. Deviation [deg]
$\phi$	-0.012	0.863
$\theta$	-0.007	0.916
$\psi$	-0.002	1.034

### 5.2.2 Results of EKF<sub>2</sub>

The second derived algorithm is going to be tested now. The main difference between this algorithm and the previous one is that it makes use of an airspeed sensor to obtain the (true) airspeed. However, this

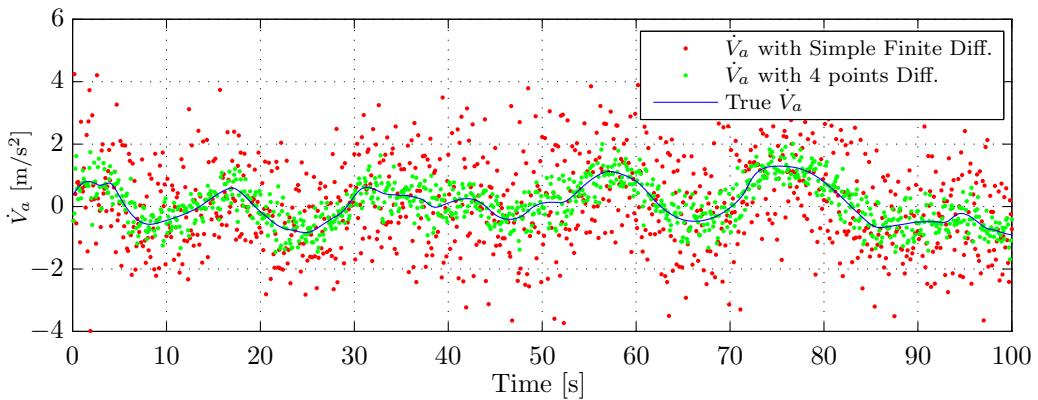
<sup>2</sup>The innovation sequence of the EKF<sub>1</sub> algorithm with Eq. (5.9) inclusion is not presented in order not to overload the results. No major conclusions would be taken from its integration here.

sensor is usually noisier than the GPS unit used to obtain the velocities through the Doppler effect. The finite differences of Eq. (4.63) highly amplify the noise which results in a less accurate estimation. Thus, some sort of filtering or differentiation over longer periods should be applied, even if it introduces some lag to the response. Let us use a smooth differentiator, that makes use of instants till  $t_{k_{GPS}-4}$  (Pavel, 2009):

$$\dot{V}_a(k_{GPS}) = \frac{V_a(k_{GPS}) + 2V_a(k_{GPS} - 1) - 2V_a(k_{GPS} - 3) - V_a(k_{GPS} - 4)}{8\Delta t} \quad (5.10)$$

where  $k_{GPS}$  is the instant of GPS measurements availability. Note that the airspeed acceleration  $\dot{V}_a$  only needs to be calculated at the same instants as the GPS availability because it is only required when the KF update step entirely runs. The comparison between the use of simple finite differences and Eq. (5.10) is presented in Figure 5.10 where we can see advantages in its implementation.

The analysis of the EKF<sub>2</sub> that follows is analogous to the one performed for the EKF<sub>1</sub> algorithm.



**Figure 5.10:** Differentiation of the noisy airspeed measurements by two methods.

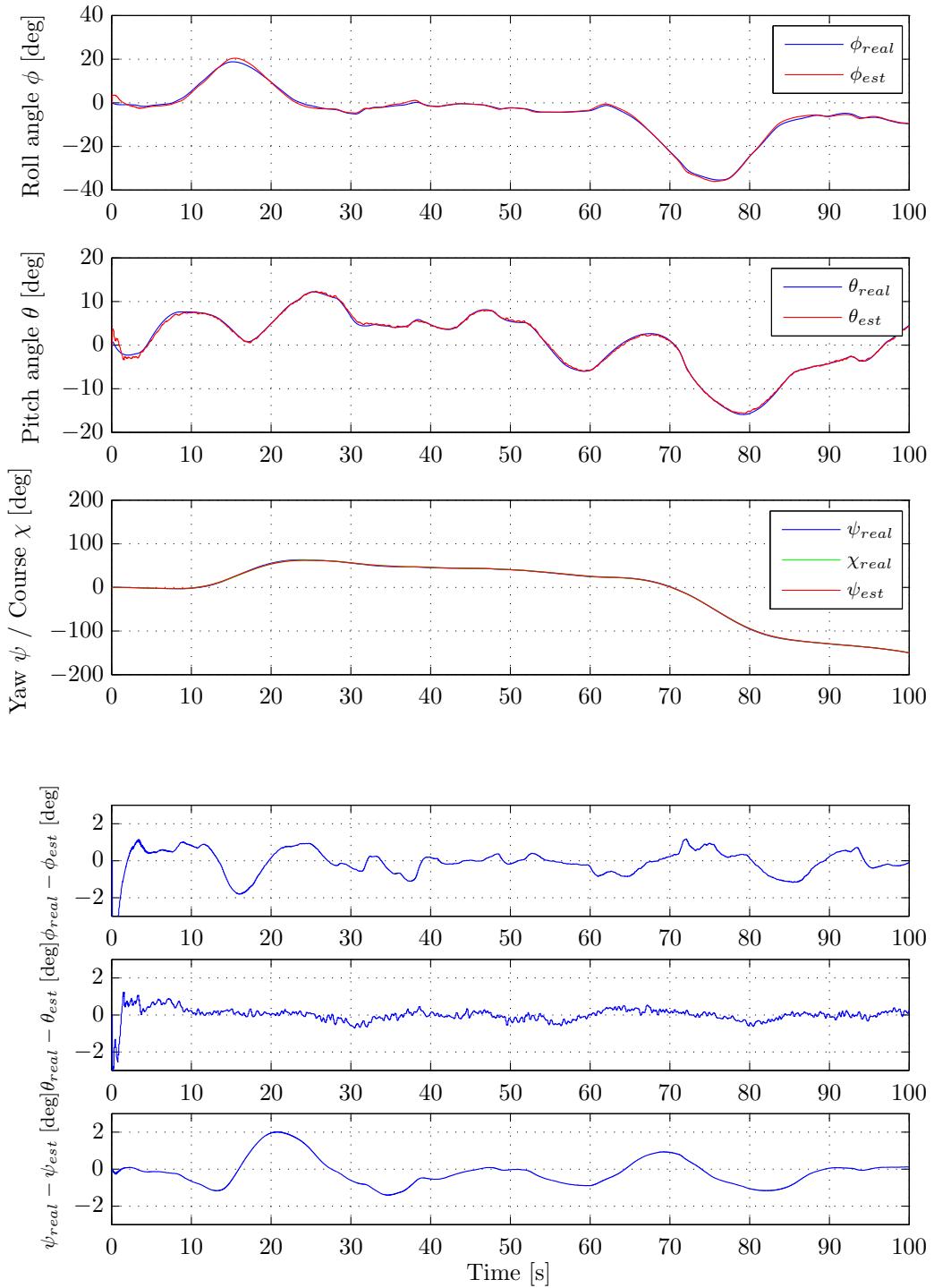
## Flight 1

The estimation results obtained with EKF<sub>2</sub> for the first flight are presented in Figures 5.11 and 5.12. It is possible to see that with the current conditions the *global* estimation error is relatively small. The numerical results with the mean error and the standard deviation can be consulted in Table 5.6.

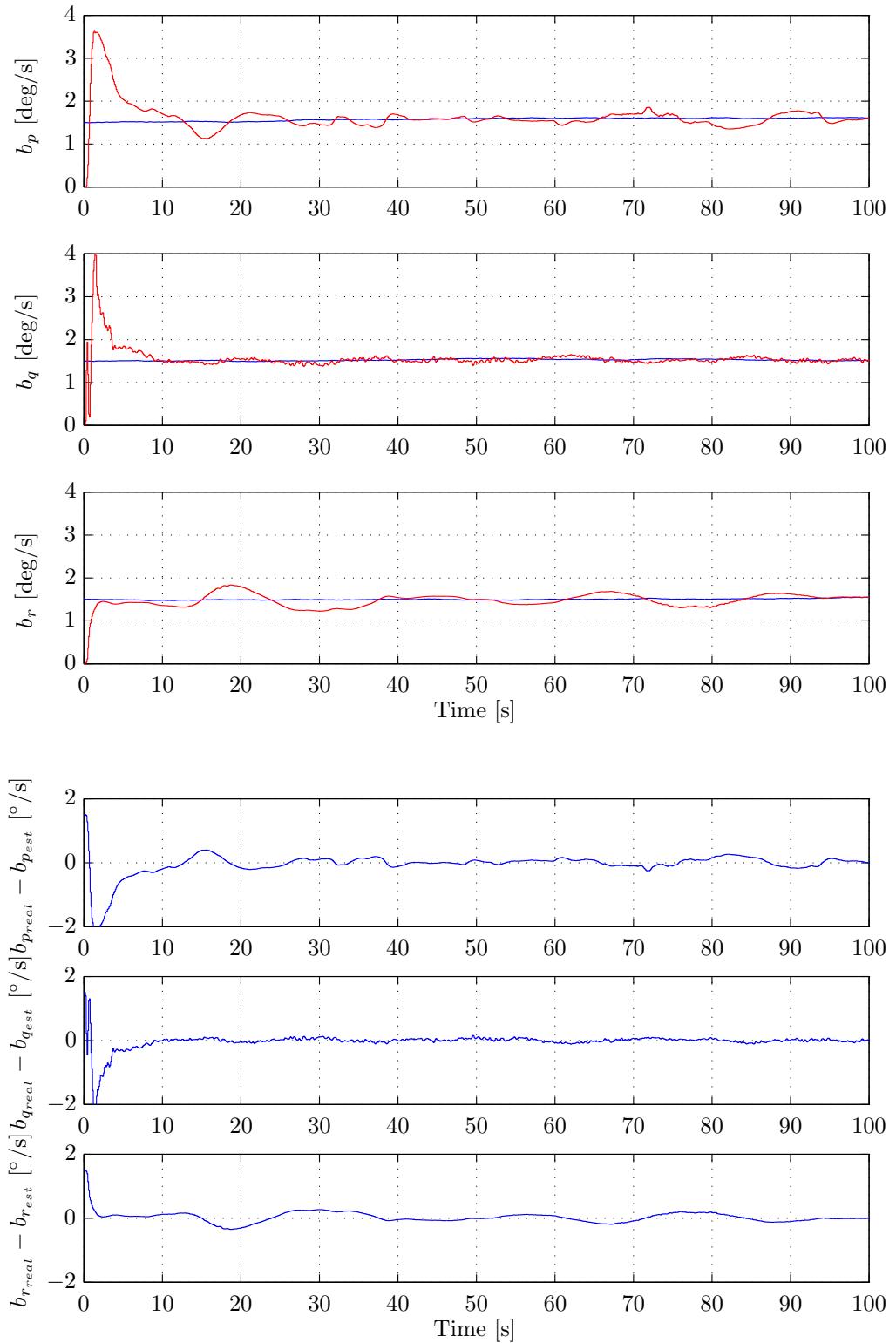
As for the attitude angles, the roll angle shows a slight decrease in estimation accuracy upon wide maneuvers when compared to the EKF<sub>1</sub> algorithm. Nevertheless, the estimation mean error is close to zero and the standard deviation is less than one degree. The same applies to the pitch angle estimation where some oscillation can be noted. This is due to the various differentiations that amplify the noise of the sensors. Furthermore, recall that this algorithm makes use of an airspeed sensor that is rather noisy and is directly used and differentiated to obtain the pitch angle measurement  $\theta_m$  (Eq. (4.60)). Finally, the yaw angle has almost the same performance as the one obtained with the EKF<sub>1</sub>.

The gyroscopes estimation algorithm take some seconds to track the *turn-on* bias but after that transient they follow the real bias with small error. Again, there is a relationship with a higher sideslip angle and a larger bias error estimation.

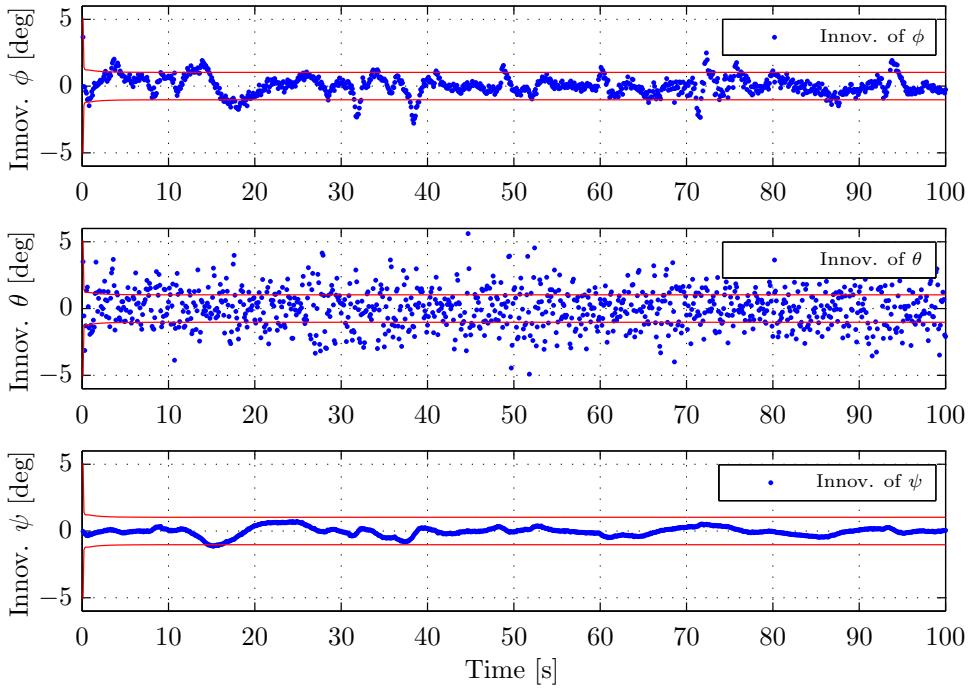
The innovation sequence of EKF<sub>2</sub> is presented in Figure 5.13 with the corresponding numerical results in Table 5.7. Although the roll and yaw innovations do not show a *white* sequence, their mean is approximately zero and are relatively inside the standard deviation of innovation. As for the pitch angle innovation, it shows a white sequence but with an increased standard deviation. Once again, it is related to the differentiations of the measurements of noisy sensors. The standard deviation of innovation for this case could be expanded by increasing the second diagonal entry of the measurements covariance matrix  $R$ . This would also slightly reduce the oscillations present in the pitch angle estimation in Figure 5.11.



**Figure 5.11:** Estimation of  $\phi$ ,  $\theta$  and  $\psi$ , and differences between real and estimated angles, with EKF<sub>2</sub>, Flight 1.



**Figure 5.12:** Estimation of  $b_p$ ,  $b_q$  and  $b_r$ , and differences between real and estimated bias with EKF<sub>2</sub>, Flight 1.



**Figure 5.13:** Innovation of the three Euler angles with EKF<sub>2</sub> for Flight 1. The red line represents the standard deviation of the Innovation.

**Table 5.6:** Numerical results of EKF<sub>2</sub> for Flight 1.

State Error	Mean	Std. Deviation
$\phi$	-0.080 deg	0.704 deg
$\theta$	0.005 deg	0.355 deg
$\psi$	-0.160 deg	0.751 deg
$b_p$	-0.046 deg/s	0.349 deg/s
$b_q$	-0.030 deg/s	0.250 deg/s
$b_r$	0.030 deg/s	0.170 deg/s

**Table 5.7:** Innovation results of EKF<sub>2</sub> for Flight 1.

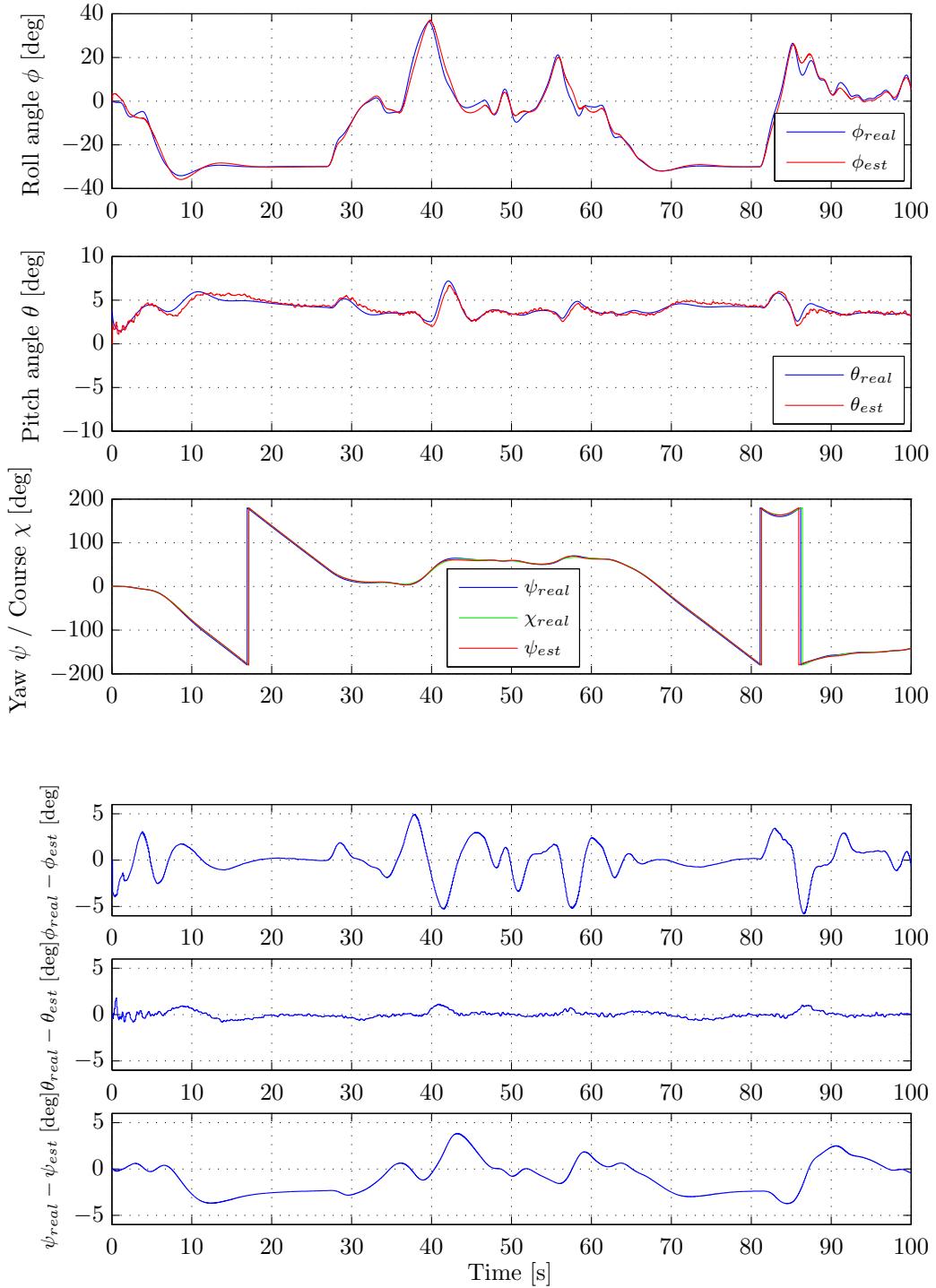
Innov. Sequence	Mean [deg]	Std. Deviation [deg]
$\phi$	0.029	0.707
$\theta$	0.018	1.454
$\psi$	-0.040	0.328

## Flight 2

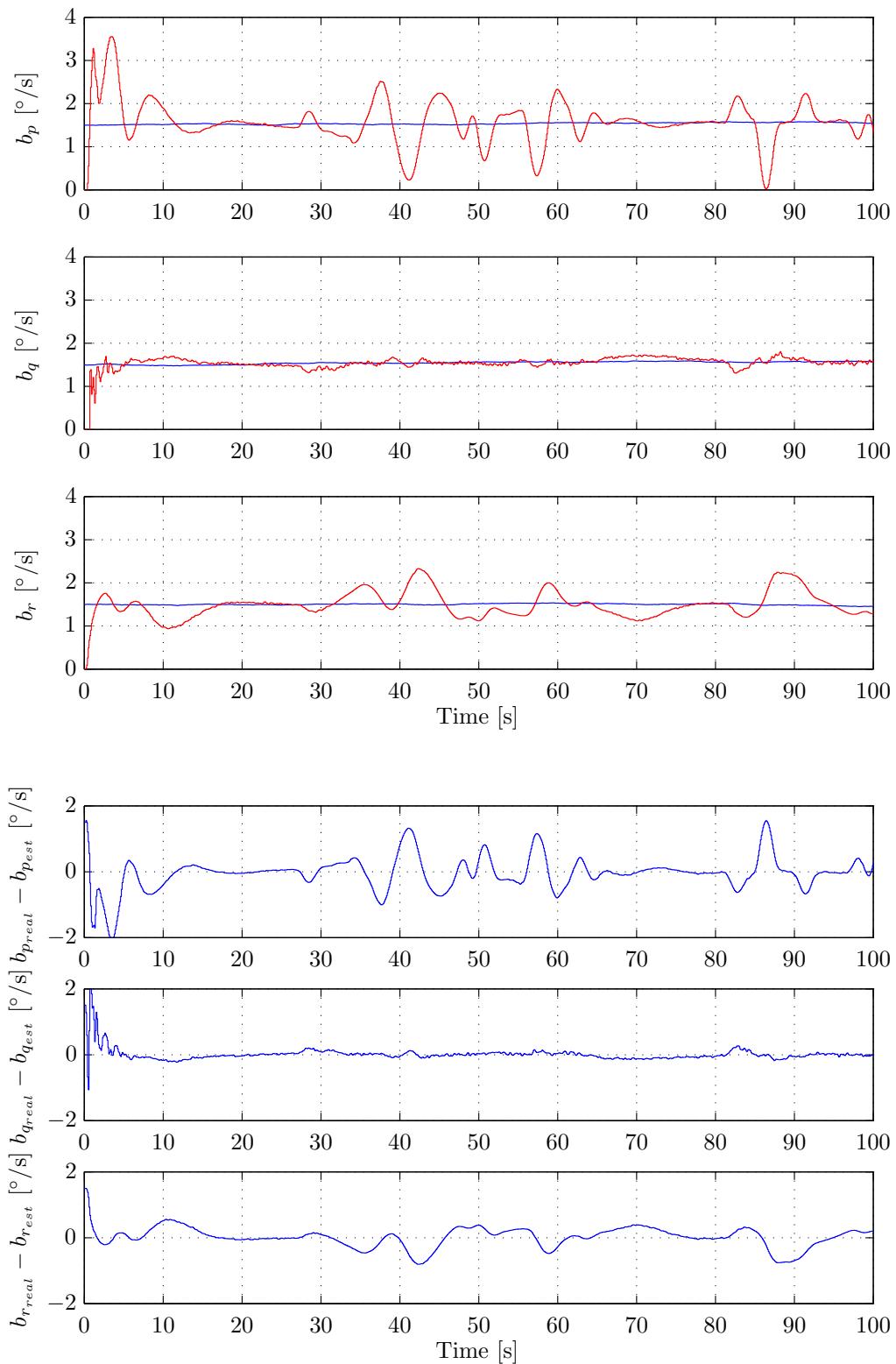
The estimation results obtained with the EKF<sub>2</sub> for the second flight are shown in Figures 5.14 and 5.15. The numerical results with the mean error and the standard deviation can be consulted in Table 5.8. As for the EKF<sub>1</sub> algorithm, we can notice an increase in the estimation error with this flight, more significant for the roll and yaw angles. Additionally, the pitch and roll angles present some sort of small lag in the beginning of new maneuvers.

Comparison between the sideslip angle in Figure C.5 and the  $b_p$  estimation in Figure 5.15 let us conclude that there is a strong relationship between them. When the sideslip angle increases, the  $b_p$  estimation also deviates from the true value. The  $b_q$  estimation is very regular and presents a very small error after the initial transient. Finally, the  $b_r$  estimation presents some errors upon wide maneuvers but it never reaches errors bigger than 0.8 degrees.

The innovation sequence can be seen in Figure D.5. It shows higher innovation values, more significant in the roll angle, with a bigger correlation, i.e., non-white sequence. The innovation values often go beyond the standard deviation of innovation which means that the entries of the  $R$  matrix are not exactly in accordance with the real measurements noise. An increase in the diagonal entries of the  $R$  matrix should



**Figure 5.14:** Estimation of  $\phi$ ,  $\theta$  and  $\psi$ , and differences between real and estimated angles, with EKF<sub>2</sub>, Flight 2.



**Figure 5.15:** Estimation of  $b_p$ ,  $b_q$  and  $b_r$ , and differences between real and estimated bias with EKF<sub>2</sub>, Flight 2.

help to minimize this problem and obtain slightly better estimation results. However, the results that would be obtained should not largely influence the estimation.

**Table 5.8:** Numerical results of EKF<sub>2</sub> for Flight 2.

State Error	Mean	Std. Deviation
$\phi$	-0.019 deg	1.753 deg
$\theta$	-0.001 deg	0.356 deg
$\psi$	-0.917 deg	1.810 deg
$b_p$	-0.041 deg/s	0.501 deg/s
$b_q$	0.011 deg/s	0.197 deg/s
$b_r$	0.015 deg/s	0.307 deg/s

**Table 5.9:** Innovation results of EKF<sub>2</sub> for Flight 2.

Innov. Sequence	Mean [deg]	Std. Deviation [deg]
$\phi$	0.047	3.285
$\theta$	-0.011	1.370
$\psi$	-0.017	1.083

### 5.2.3 GPS outages

The advantages of a GPS/INS integration generally come into play when GPS signals are blocked or subject to degradation. To evaluate the performance of both developed EKF during these situations, a GPS gap was introduced in the simulation data of the Flight 1. It was chosen to have a duration of 20 seconds (starting at  $t = 60$ s and finishing at  $t = 80$ s). During this period, the UAV is performing a turn to the left and pitching down after  $t = 70$ s.

The estimation results for both EKF<sub>1</sub> and EKF<sub>2</sub> are presented in Figures D.6 and D.7. In Figure 5.16, the difference between the real and the estimated angles is showed for a clearer understanding of the estimation accuracy with no GPS signals. Note that the time range of the Flight 1 that we can see in Figure 5.16 was shorten to  $t \in [50, 100]$ s without loss of generality.

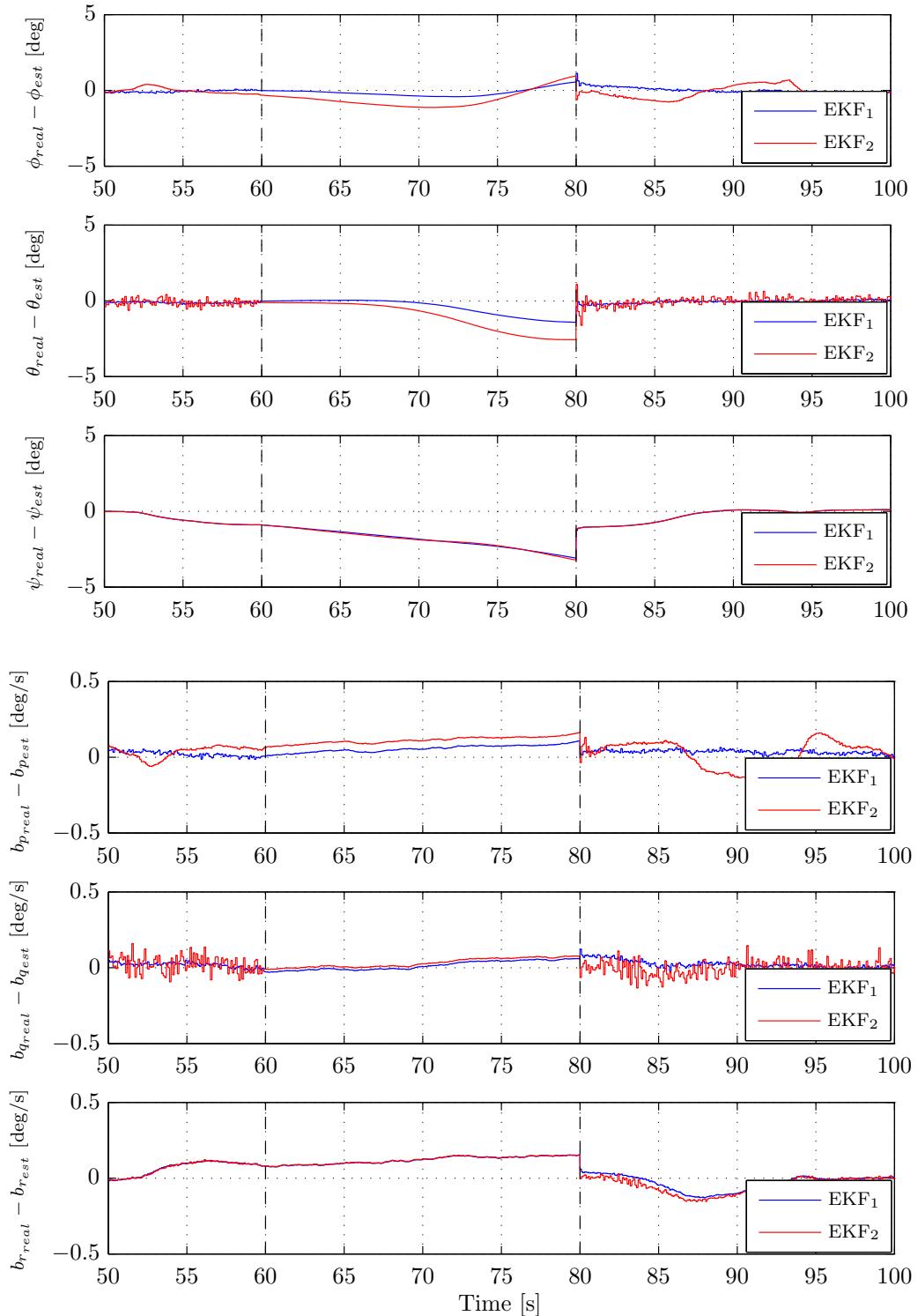
As it is possible to see, the performance of both EKF is similar; especially for the bias estimation they have exactly the same behavior (but with different final estimation errors). This happens because the two algorithms make use of the same Process model, with equal sensors' characteristics and process/measurement covariance matrices. Additionally, the inherent characteristics of the EKF<sub>1</sub> and EKF<sub>2</sub> algorithms result in the same error covariance matrix  $P$ . Thus, the difference between the results is related to the value of the states at  $t = 60$ s. If the bias of the gyroscopes is well estimated at that instant, the estimation during the GPS gap is more accurate.

The numerical characteristics of the estimation, exactly before the GPS signal becomes available again, are present in Table 5.10.

**Table 5.10:** Deviation of the estimated states from the real states at  $t = 79.99$ s.

EKF <sub>1</sub>		EKF <sub>2</sub>	
State	Error	State	Error
$\phi$	0.557 deg	$\phi$	0.959 deg
$\theta$	-1.412 deg	$\theta$	-2.555 deg
$\psi$	-3.081 deg	$\psi$	-3.212 deg
$b_p$	0.106 deg/s	$b_p$	0.161 deg/s
$b_q$	0.060 deg/s	$b_q$	0.077 deg/s
$b_r$	0.155 deg/s	$b_r$	0.157 deg/s

The overall performance of the estimation is relatively good with the bias of the gyroscopes diverging only some tenths of degree. Concerning all the attitude states, the EKF<sub>1</sub> reveals a better accuracy, especially for the pitch angle estimation. After the GPS unavailability, the estimation of the states quickly returns to the previous performance (presented in the last two sections) of the EKF<sub>1</sub> and EKF<sub>2</sub>.



**Figure 5.16:** States estimation upon GPS outages during  $t \in [60, 80]$ s (Flight 1).

#### 5.2.4 Simulation Summary

In this chapter both  $\text{EKF}_1$  and  $\text{EKF}_2$  were simulated in order to test their performance. Two flights were conducted, the first representing a general flight and the second was conducted in order to verify the limits of the estimation algorithms. In the latter case, the flight characteristics assumptions, adopted in Chapter 4, were somewhat disregarded and the performance of the estimation accuracy evaluated. In the last part of the simulation, GPS outages were introduced to check the performance when only the inertial navigation system is working.

Overall, the  $\text{EKF}_1$  achieved a better performance than the  $\text{EKF}_2$ , but both algorithms resulted in relatively small estimation errors, as can be seen from Tables 5.2, 5.4, 5.6 and 5.8.

As expected, the sideslip angle influence the yaw angle estimation accuracy since it was assumed to be zero upon the derivation of the estimation algorithms. Nevertheless, the algorithms behave relatively well during the presence of small sideslip angles, and can very accurately track the real yaw value after it returns close to zero. As for the roll angle estimation, its performance is rather regular and no major problems arise when the UAV reaches high roll values. Finally, the pitch angle estimation required more attention, especially with the  $\text{EKF}_1$  algorithm. Nevertheless, it was found that with the inclusion of Eq. (5.9) in this algorithm, the error offset in the pitch angle estimation of Figure 5.8 could be solved almost completely, improving the estimation accuracy.

The three different bias estimations,  $b_p$ ,  $b_q$  and  $b_r$ , converge to the turn-on bias and then track the oscillating (Gauss-Markov process) true biases values. A relation also exists between the  $b_r$  estimation error and the sideslip. In the presence of a sideslip angle, it is *seen* as a bias, and the estimation error increases (for instance compare Figure C.5 and Figure 5.9).

The reliability testing of the two derived EKF was also performed. All the innovation sequences demonstrated a mean around zero, mainly within the standard deviation of the innovation. For the  $\text{EKF}_2$  this could have been improved by changing the entries of the measurements covariance matrix  $R$  (recall that the  $\text{EKF}_2$  makes uses of noisier sensors with a higher number of numerical differentiation). Additionally, the roll and pitch innovation showed a more white behavior than the yaw angle, that was somewhat correlated (relationship between the innovation of the yaw and the sideslip angle can also be tracked by comparing e.g, Figures D.4 and C.5).

During GPS outages, the two algorithms have almost the same performance because both make use of the same Process model with equal sensors and noise covariance matrices characteristics. It was concluded that the attitude of the UAV can still be tracked upon periods of GPS unavailability.

### 5.3 Real Flight

The application to real flight data is the ultimate test of an algorithm. In this chapter, the  $\text{EKF}_1$  algorithm is going to be tested with a real flight. The  $\text{EKF}_2$  algorithm could not be tested due to the absence of real flight data with measurements from an airspeed sensor. However, the focus of this thesis was mainly in an GPS/INS integration with the usage of GPS-derived accelerations which also showed better results in the previous section.

The device used to collect the real data was a *MTi-G Xsens* device, (Xsens Technologies, 2009). More information about this device is presented in Appendix E.1. It can provide us the *raw* data<sup>3</sup> necessary to apply the  $\text{EKF}_1$  algorithm. In addition, the MTi-G also estimates various position, velocity and attitude parameters, which includes the roll, pitch and yaw angles. Thus, it will be possible to notice if the device MTi-G and our algorithm  $\text{EKF}_1$  estimated Euler angles are somewhat in accordance. The flight was performed on 2010-10-13 at Delft, The Netherlands, and a post-processing of the data was conducted.

<sup>3</sup>The *real* raw data that the device says it provides is somewhat doubtful and we do not really know if any kind of simple filtering is firstly applied. No additional information was found in the manual.

The utilized UAV is highly dynamic, performing wide maneuvers in few seconds. Figure 5.17 shows the *Easystar* UAV, used to obtain the flight data.



**Figure 5.17:** Easystar UAV with 1.4m wingspan and 0.68kg of (empty) weight.

The real flight implementation raises problems that will be discussed in Section 5.3.1. Then, in Section 5.3.2 the  $\text{EKF}_1$  algorithm is going to be tested with real flight data. Its reliability is tested and comparison between the estimated attitude angles obtained with  $\text{EKF}_1$  and the ones estimated by the MTi-G device is done.

### 5.3.1 Real Flight Issues

The lack of information about the inertial sensors of the MTi-G Xsens device makes the modeling of the sensors more difficult. However, as presented in Section 4.1.1 this issue can be overcome by assuming the bias of the gyroscopes as:

$$\dot{\underline{b}} = \underline{\varepsilon}_b \quad (5.11)$$

where the noise term  $\underline{\varepsilon}_b$  incorporates the *full noise* of the bias-drift. The variance of this white Gaussian noise parameter is then adjusted manually till reasonable results are obtained. The KF's continuous-time state-space system of the process model becomes:

$$\underline{f}[\underline{x}(t), \underline{u}(t), t] = \begin{bmatrix} p_m - b_p + (q_m - b_q) \sin \phi \tan \theta + (r_m - b_r) \cos \phi \tan \theta \\ (q_m - b_q) \cos \phi - (r_m - b_r) \sin \phi \\ (q_m - b_q) \frac{\sin \phi}{\cos \theta} + (r_m - b_r) \frac{\cos \phi}{\cos \theta} \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (5.12)$$

The noise and bias characteristics are presented in Table E.1. Other considerations that should be referred are:

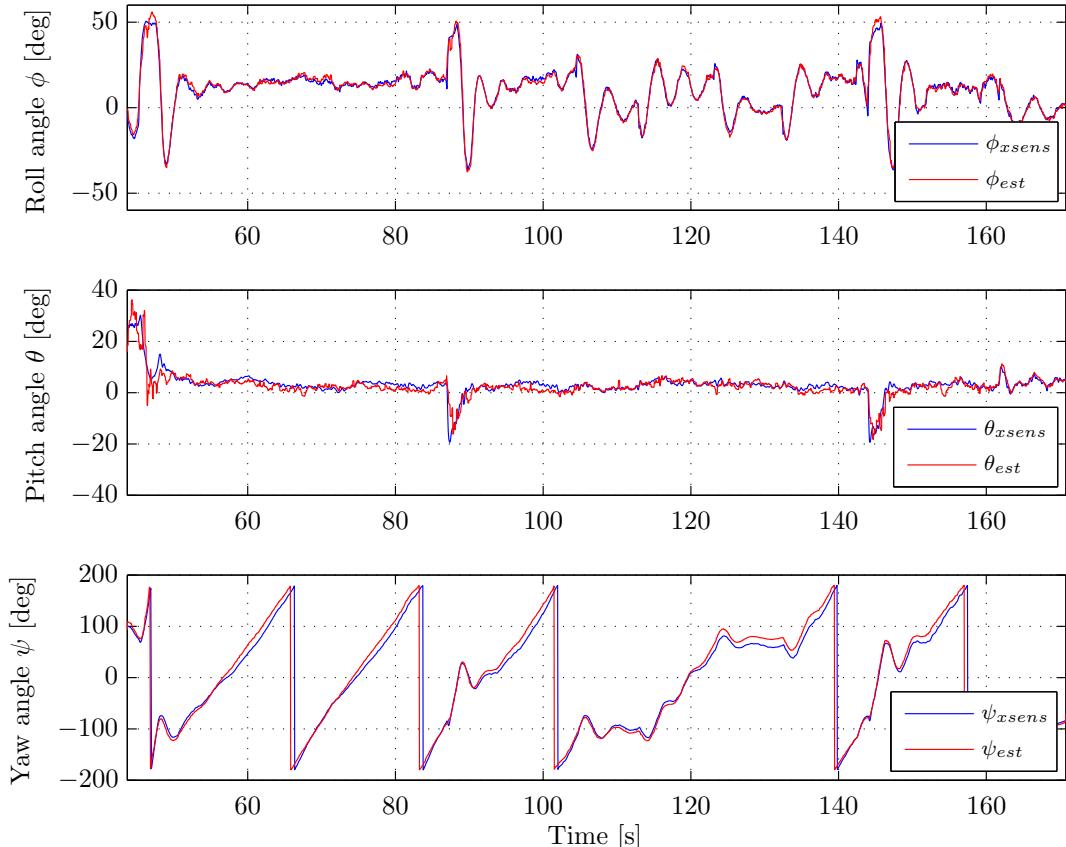
- The  $\text{EKF}_1$  is evaluated in post-processing, where all the data are aligned to GPS-time.
- The GPS/INS synchronization problem was already discussed in Section 5.1.2. The 1PPS GPS signal was utilized to synchronize both navigation systems.
- The MTi-G device needs some start-up time to obtain a GPS-fix and accurately provide velocities estimates. Therefore, the first velocities estimates should be handled carefully.
- With the default configuration, the MTi-G outputs the data of the inertial sensors at a frequency of 100Hz and GPS measurements at 4Hz.

- During the post-processing, the verification of each recorded line is performed, where each line is scanned to see if all recorded bites are correct and can be accounted. Although the manual of the device says that the flight data is recorded at 100Hz, it turned out that not every recorded line could be used. The post processing resulted in about 25% of useful recorded data lines which made the synchronization more difficult.
- Attention should be paid to the GPS velocity magnitude. If it is around zero the heading might become unobservable and no reliable KF update can be performed.

### 5.3.2 Real Flight Results

The real flight consists in highly dynamic maneuvers performed by a small UAV constituted mainly by 360 degrees turns at constant altitude (see Figure E.1). The GPS velocity (groundspeed) of the aircraft varies approximately between 12 and 18m/s (see Figure E.2). The measurements of the accelerometers and gyroscopes can be seen in Figure E.3. All the relevant flight data information is presented in Appendix E where it is also possible to find the definition of the initial states estimation  $\hat{x}(0|0)$ , the initial error covariance matrix  $P(0|0)$ , the parameter  $\lambda$ , and the noise covariance matrices  $Q$  and  $R$  utilized. The estimation algorithm runs while the aircraft is in the air ( $t \in [44.12, 171.22]$ s) and no GPS outages were found.

The estimated attitude angles, roll  $\phi$ , pitch  $\theta$  and yaw  $\psi$  can be seen in Figure 5.18. Since no information about the *real* attitude angles is available, the estimated Euler angles provided by the MTi-G Xsens device is also included in Figure 5.18 for a matter of comparison.



**Figure 5.18:** Estimation of the attitude angles  $\phi$ ,  $\theta$  and  $\psi$  for the real flight.

We can see a consistent attitude tracking for all the angles. The roll angle is highly dynamic where we can notice wide maneuvers of almost 90 degrees (from  $\phi \approx 50\text{deg}$  to  $\phi \approx -40\text{deg}$ ) in just 1.6 seconds (check  $t \approx 88\text{s}$  or  $t \approx 166\text{s}$ ). Nevertheless, the algorithm seems to accomplish this fast attitude estimation without major difficulties. The pitch angle is consistent with an almost horizontal flight with two fast *pitch down* at  $t = 87\text{s}$  and  $t = 144\text{s}$ . The yaw angle estimation shows that the UAV performed several complete turns. We can also notice the fast initial convergence of the EKF. The estimation of the various bias of the gyroscopes is presented in Figure E.4 where it is possible to see an initial oscillation before a convergence to an almost constant value probably representing the constant bias of the gyroscopes.

In Figure 5.18, we can additionally observe the estimated angles done by the MTi-G Xsens device. Comparison between the two estimations let us conclude that they present small deviations between each other. Nevertheless, the yaw angle seems to exhibit some kind of *offset*. Since the MTi-G unit uses more instruments to estimate the attitude (such as magnetometers) it might be related to the sideslip angle. Small and light UAVs like this one can fly during long periods with high sideslip angles. However, since no true states are available we can not know which estimation algorithm is closer to the true yaw angle. In Table 5.11 the differences between the two estimations are presented<sup>4</sup>. It is noticeable the similar performances of the roll and pitch estimations.

**Table 5.11:** Numerical results of Real Flight comparing with the MTi-G device.

State Error	Mean [deg]	Std. Deviation [deg]
$\phi$	-0.118	2.083
$\theta$	0.547	2.211
$\psi$	-5.788	8.273

The best way to evaluate the performance of the EKF when no *true* states are available is by assessing the innovation sequences. Figure 5.19 shows the innovation sequences and Table 5.12 their numerical results. They present an average close to zero and standard deviations around 4 degrees. Additionally, the innovation sequences have a similar behavior to the ones obtained in the simulation: the roll and pitch estimations present a more *white* sequence and the yaw a more correlated one.

**Table 5.12:** Innovation results of Real Flight.

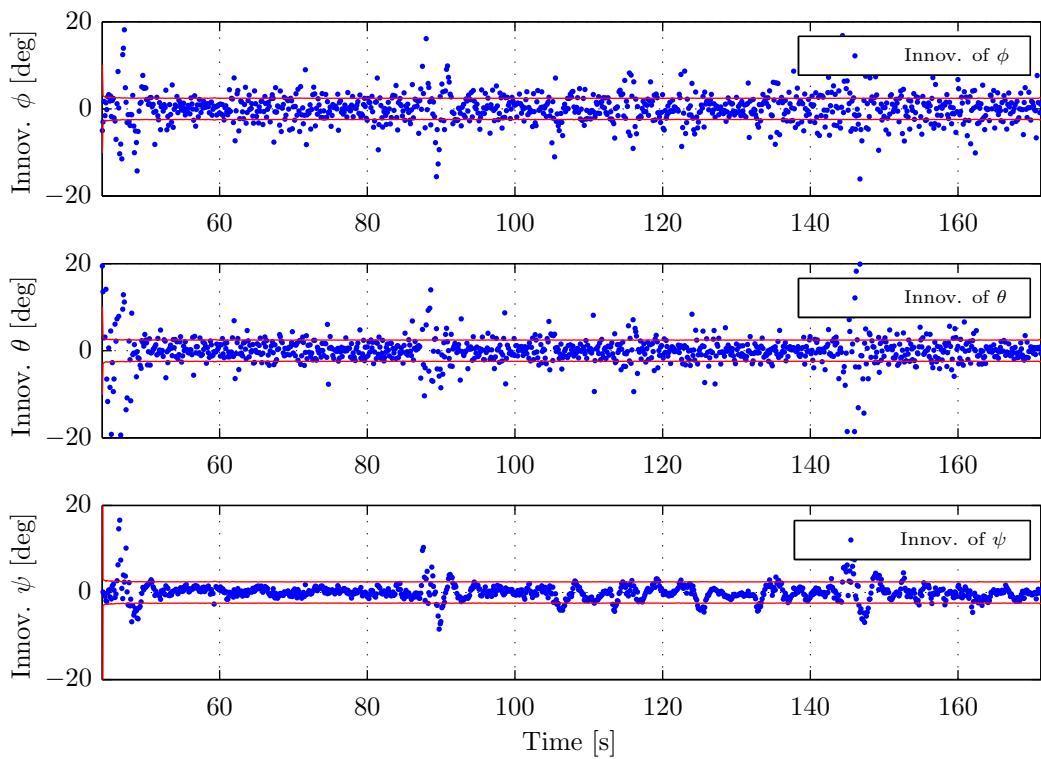
Innov. Sequence	Mean [deg]	Std. Deviation [deg]
$\phi$	0.046	3.636
$\theta$	0.033	4.139
$\psi$	0.109	3.928

As a matter of curiosity, a particular aspect of the EKF<sub>1</sub> algorithm is its estimation speed<sup>5</sup>. The time required to estimate the three attitude angles and the bias of the gyroscopes for the complete flight ( $t_{\text{flight}} = 127.10\text{s}$ ) was only 0.569 seconds. The time average of each iteration was  $t = 0.122\text{ms}$  without GPS update and  $t = 0.213\text{ms}$  with measurements update.

---

<sup>4</sup>The MTi-G device does not provide the estimation of the bias of the gyroscopes, nor numerical results of the innovation sequences.

<sup>5</sup>The computer specifications where the EKF<sub>1</sub> algorithm ran are: Intel Core<sup>TM</sup> 2 Duo CPU T9550 2.66GHz with 4Gb RAM-Memory and Windows Vista Operating System.



**Figure 5.19:** Innovation of the three Euler angles for the real flight. The red line represents the standard deviation of the Innovation.

# Chapter 6

## Conclusions and Recommendations

The concluding remarks of this thesis and recommendations for future work are presented in this chapter.

### 6.1 Conclusions

The work developed in this thesis concerned the derivation of a GPS/MEMS-INS (GPS/MEMS-AHRS) integration algorithms that would provide a fast, efficient and reliable attitude estimation for small highly dynamic UAVs. A loosely coupled integration architecture was chosen due to its simple implementation and operational speed, avoiding hardware handling and state augmentation. The main conclusions that can be drawn from this thesis are the following:

- Extensive sensor research was performed that led to a complete accelerometers and gyroscopes description. This could have been applied more deeply if the MEMS sensors were better documented.
- It was possible to arrive to analytical solutions for the measurements models of the derived EKF. This resulted in two *Half* Extended Kalman filters with equal nonlinear process model and distinct linear measurements models. One of the derived EKF had a measurements model focused in GPS accelerations, and the second derived EKF in the usage of an airspeed sensor. It was verified that the undergoing of the nonlinear calculations/transformations keep the noise distribution relatively Gaussian.
- Concerning the simulation results, two flights were performed to test the two derived attitude determination algorithms. Overall, the  $EKF_1$  performed better than the  $EKF_2$ , but both algorithms resulted in relatively small estimation errors. The best attitude estimation was attained for the roll and pitch angles. The yaw angle accuracy was confirmed to degrade upon flight situations where sideslip was present. However, this increased inaccuracy was relatively small and lasted for short periods. The several differentiations present in the  $EKF_2$  amplified the noise of the sensors and made the pitch angle estimation more oscillating. These differentiations could be considered a drawback when compared with the  $EKF_1$  algorithm and this would be a sufficient reason to prefer the  $EKF_1$  rather than the  $EKF_2$ . Additionally, the airspeed sensor is a more noisy sensor along with the fact that it might be relatively difficult to integrate this sensor in the UAV/MAV's frame.
- The simulation of the  $EKF_1$  algorithm for the second flight resulted in a constant error in the pitch angle estimation. It was found out that this could be solved with a relation between the angle of attack and the roll angle. Integration of this relation in the architecture of the  $EKF_1$  led to a better estimation accuracy for both pitch and yaw angles. Overall, even in more demanding situations that went beyond the algorithm assumptions, the estimation accuracy was not heavily decreased.

- The bias estimation tracked the constant turn-on bias introduced in the simulated gyroscopes, and approximately followed the bias of the gyroscopes, modeled as a first-order Gauss-Markov process. The estimation error increased upon certain highly dynamic maneuvers where the zero sideslip assumption dropped. In these situations the sideslip angle was *seen* as a gyroscope's bias and the estimation error rose. However, these errors were relatively small and the estimation rapidly came back close to the true bias of the gyroscope.
- Besides the comparison of the estimated angles with the real angles, the innovation sequences of both algorithms were verified to test their reliability. All the innovation sequences demonstrated a mean close to zero, mainly within the standard deviation of the innovation. In addition, the roll and pitch innovations presented a more white behavior than the yaw angle, that was somewhat correlated to the sideslip angle behavior.
- GPS outages were introduced to verify the algorithms performance when only the Inertial System was providing measurements. Results showed that for small outages of 20 seconds the performance does not degrade so quickly, with the specified characteristics of the sensors, and the attitude of the UAV can still be tracked. The estimation results in these situations show that the performance of both algorithms is very similar because they make use of the same process model.
- The EKF<sub>1</sub> algorithm was tested with real UAV flight data in a post-processing method. The real data showed that the UAV performed highly dynamic maneuvers in just some tenths of second taking the estimation algorithm to the limit. The EKF<sub>1</sub> algorithm demonstrated a consistent attitude tracking for all the attitude angles. Comparison of EKF<sub>1</sub> attitude estimation with the one provided by the MTi-G Xsens device showed that they are mainly in accordance. The biggest discrepancy can be found with the yaw angle estimation that may be caused by high sideslip angles (the MTi-G device also uses additional aiding sensors to obtain an attitude estimation). Nevertheless, reliability testing with the innovation sequences show that the EKF<sub>1</sub> is working properly and presents the same behavior as in the simulations.
- As a final remark, the algorithms revealed a very short processing time, completing the stated objectives of obtaining a simple, efficient, low-cost and low-power demanding GPS/INS integration algorithm(s).

## 6.2 Recommendations

The following topics are recommended for future work in GPS/INS-MEMS (GPS/AHRS-MEMS) integration research:

- *Test of the EKF<sub>2</sub> algorithm with real UAV flight data.* As explained before, the EKF<sub>2</sub> algorithm was not tested with real flight data due to the unavailability of data with airspeed sensor measurements. Applying this algorithm to real data would make the comparison of both derived architectures possible.
- *Estimate the sideslip angle.* The sideslip angle is the one of the main factors in the estimation inaccuracies. Obtaining its value would complement the estimation algorithm and turn the accuracy even higher and more reliable in situations where this angle becomes not negligible. Differentiation between course and yaw angle would then be possible. Even with the main difficulties referred about the integration of magnetometers in UAVs, if this device is well calibrated and the magnetic interferences reduced it could turn out to be a decent aiding sensor.
- *Wind estimation.* The wind can make the UAV's heading unobservable or induce high sideslip angles. Its estimation would help to solve this problems and improve accuracy.

- *Adaptive Kalman Filter.* It might be possible to improve the estimation accuracy during highly dynamic maneuvers or GPS reduced accuracy with varying noise covariance matrices.
- *Quaternions attitude representation.* It was possible to confirm that the pitch angle was kept far from the 90 degrees singularity. However, it is known that the highly dynamic UAVs can reach this value. Thus, research about quaternions attitude representation could also be done.
- *Introduction of more states.* Integration of the developed AHRS/GPS algorithm with an expanded state algorithm including position and velocity estimates could be done (several literature already exists). Overall, the state augmentation should be balanced at the expense of a higher complexity and power requirements.

# Bibliography

- Abdel-Hafez, M. (2009). On the Development of an Inertial Navigation Error-budget System. In *Journal of the Franklin Institute* (Vol. 348, p. 24 - 44). Elsevier.
- Ascher, U. M., & Petzold, L. R. (1998). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations* (Vol. 61). Society for Industrial Mathematics.
- Babu, R., & Wang, J. (2005, December). Ultra-Tight GPS/INS/PL integration: Kalman Filter Performance Analysis. In *GNSS 2005* (p. 8 - 10). Hong Kong.
- Barbour, N. M. (2004, 31 May - 1 June). Inertial Navigation Sensors. *NATO RTO Lecture Series, 232*.
- Brogan, W. L. (1974). *Modern Control Theory*. Quantum Publishers.
- Brown, A., & Lu, Y. (2004, September). Performance Test Results of an Integrated GPS/MEMS Inertial Navigation Package. In *Proceedings of ION GNSS 2004* (p. 825 - 832).
- Brown, R., & Hwang, P. (1997). *Introduction to Random Signals and Applied Kalman Filtering*. New York: Wiley Interscience.
- Caron, F., Duflos, E., Pomorski, D., & Vanheeghe, P. (2006). GPS/IMU Data Fusion Using Multisensor Kalman Filtering: Introduction of Contextual Aspects. *Information Fusion*, 7(2), 221 - 230.
- Chalko, T. J. (2007). *High accuracy speed measurement using GPS (Global Positioning System)* (Tech. Rep.). Mount Best, Victoria, Australia: Scientific Engineering Research P/L.
- Cheng, L., Zhaoying, Z., & Xu, F. (2008, October). Attitude Determination for MAVs Using a Kalman Filter. *Tsinghua Science and Technology*, 13(5), 593 - 597.
- Chiang, Y., Wang, L., Chang, F., & Peng, H. (2002). Constrained filtering method for attitude determination using GPS and gyro. In *IEEE Proceedings Radar, Sonar and Navigation* (Vol. 149, p. 258 - 264).
- Chu, Q. P. (2011). *Spacecraft Attitude Dynamics and Control - Course Notes 2010/2011*. (Delft University of Technology)
- Ding, W., Wang, J., Li, Y., Mumford, P., & Rizos, C. (2008, February). Time Synchronization Error and Calibration in Integrated GPS/INS Systems. *Electronics and Telecommunications Research Institute (ETRI)*, 30(12), 59 - 67.
- Eldredge, A. M. (2006). *Improved State Estimation for Miniature Air Vehicles*. Unpublished master's thesis, Brigham Young University.
- El-Rabbany, A. (2002). *Introduction to GPS: the Global Positioning System* (First, Ed.). Boston, MA: Artech House, Inc.
- Etkin, B., & Reid, L. D. (1996). *Dynamics of Flight: Stability and Control*. John Wiley & Sons, Inc.
- Gebre-Egziabher, D., & Elkaim, G. (2008). MAV attitude determination by vector matching. *Aerospace and Electronic Systems, IEEE Transactions on*, 44(3), 1012 - 1028.
- Gebre-Egziabher, D., Hayward, R., & Powell, J. (2004). Design of Multi-sensor Attitude Determination Systems. *Aerospace and Electronic Systems, IEEE Transactions on*, 40(2), 627 - 649.
- Gebre-Egziabher, D., Hayward, R. C., & Powell, J. D. (1998). A Low Cost GPS/Inertial Attitude

- Heading Reference System (AHRS) for General Aviation Applications. In *Proceedings of the IEEE Position Location and Navigation Symposium, PLANS 1998* (p. 518 - 525).
- Gebre-Egziabher, D., Klkaim, G. H., Powell, J., & Parkinson, B. W. (2000, March). A Gyro-Free Quaternion-Based Attitude Determination System Suitable for Implementation Using Low Cost Sensors. In *Proceedings of the IEEE Position, Location, and Navigation Symposium* (p. 185 - 192).
- George, M., & Sukkarieh, S. (2005). Tightly Coupled INS/GPS with Bias Estimation for UAV Applications. In *Proceedings of Australasian Conference on Robotics and Automation*.
- Gleason, S., & Gebre-Egziabher, D. (2009). *GNSS Applications and Methods*. Artech House Publishers.
- Godha, S. (2006). *Performance Evaluation of Low Cost MEMS-Based IMU Integrated with GPS for Land Vehicle Navigation Application*. Unpublished master's thesis, The University of Calgary, Canada.
- Grewal, M. S., & Andrews, A. P. (2001). *Kalman Filtering: Theory and Practice Using MATLAB* (Second ed.). New York, NY: John Wiley & Sons, Inc.
- Grewal, M. S., Weill, L. R., & Andrews, A. P. (2001). *Global Positioning Systems, Inertial Navigation, and Integration* (First ed.). John Wiley & Sons Inc.
- Hayward, R., Gebre-Egziabher, D., Schwall, M., Powell, J., & Wilson, J. (1997). Inertially Aided GPS Based Attitude Heading Reference System for General Aviation Aircraft. In *Proceedings of the Institute of Navigation ION-GPS Conference* (p. 1415 - 1424).
- Hummelink, B. A. (2011). *Fixed-Wing UAV Integrated Navigation with Low-Cost IMU/GPS*. Unpublished master's thesis, Delft University of Technology.
- Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. New York: Academic Press.
- Kalman, R. (1960, March). A New Approach to Linear Filtering and Prediction Problems. In *ASME Journal of Basic Engineering* (Vol. 82, p. 35 - 45).
- Kaplan, E. D., & Hegarty, C. (2006). *Understanding GPS Principles and Applications* (Second ed.). Artech House, Boston, MA.
- Kayton, M., & Freid, W. R. (1997). *Avionics Navigation Systems* (Second ed.). Wiley Interscience.
- Kim, J., Sukkarieh, S., & Wishart, S. (2006). Real-time Navigation, Guidance, and Control of a UAV using Low-cost Sensors. In *Field and Service Robotics* (Vol. 7, p. 299 - 309).
- Kingston, D. B. (2004). *Implementation Issues of Real-time Trajectory Generation on Small UAVs*. Unpublished master's thesis, Brigham Young University.
- Kornfeld, R. P., Hansman, R. J., & Deyst, J. J. (1998). Single-antenna GPS-based aircraft attitude determination. *Navigation*, 45(1), 51 - 60.
- Lewantowicz, A. H. (1992). Architectures and GPS/INS Integration: Impact on Mission Accomplishment. *IEEE Position, Location and Navigation Symposium*, 7(6), 284 - 289.
- Li, D., & Wang, J. (2005, 8-10 December). Enhancing the Performance of Ultra-Tight Integration of GPS/PL/INS: A Federated Filter Approach. *Journal of Global Positioning Systems*, 5(1 - 2), 96 - 104.
- Li, Y., Dempster, A., Li, B., Wang, J., & Rizos, C. (2006). A Low-cost Attitude Heading Reference System by Combination of GPS and Magnetometers and MEMS Inertial Sensors for Mobile Applications. In *Journal of Global Positioning Systems* (Vol. 5, p. 88 - 95).
- Lightsey, E., & Madsen, J. (2003). Three Axis Attitude Determination Using GPS Signal to Noise Ratio Measurements. *AIAA Journal of Guidance, Control, and Dynamics*, 26, 304 - 310.
- Machado, F. M. R. (2011). *UAV State Determination based on the MEMS-IMU/GPS Loosely Coupled Integration*. Unpublished master's thesis, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal.
- Maybeck, P. S. (1994). *Stochastic Models, Estimation, and Control* (Vol. 1). Navtech Book & Software Store.
- Metni, N., Pflimlin, J.-M., Hamel, T., & Soueeres, P. (2005, August). Attitude and Gyro Bias Estimation for a Flying UAV. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (p. 295

- 301).
- Mulder, J., Staveren, W. H. J. J. van, & Vaart, J. C. van der. (2011). *Flight Dynamics - Lecture Notes 2010/2011*. (Delft University of Technology)
- Mulder, M. (2010). *Avionics and Operations Course, Lecture Notes*. (Delft University of Technology)
- Nunes, F. D. (2011). *Air Traffic Control Systems - Lecture Notes*. (Instituto Superior Tecnico, Lisboa)
- Pallett, E. (1996). *Aircraft Instruments and Integrated Systems* (Second ed.). Addison Wesley Longman Ltd.
- Park, M. (2004). *Error Analysis and Stochastic Modelling of MEMS Based Inertial Sensors for Land Vehicle Navigation Applications*. Unpublished master's thesis, Dept. of Geomatics Eng., University of Calgary, Calgary, Canada.
- Pavel, H. (2009, August). *One-Sided Differentiators* (Tech. Rep.). Holoborodko.
- Persa, S. F. (2006). *Sensor Fusion in Head Pose Tracking for Augmented Reality*. Unpublished doctoral dissertation, Delft University of Technology.
- Petovello, M. G. (2003). *Real-time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation*. Unpublished doctoral dissertation, The University of Calgary, Canada.
- Pinder, S. D., Crowe, T. G., & Nikiforuk, P. N. (2001). Application of the Global Positioning System in Determination of Vehicular Acceleration. In *AIAA JAircraft* (Vol. 38, p. 856 - 859).
- Premerlani, W., & Bizard, P. (2009). *Direction Cosine Matrix IMU: Theory*. (Available: <http://gentlenav.googlecode.com/files/DCMDraft2.pdf>)
- Psiaki, M. L. (2006, March). Global Navigation Satellite Systems: Genesis, State of the Art, and Future Directions. In *Plenary Presentation at the 46th Israel Annual Conference on Aerospace Sciences*. Tel Aviv and Haifa, Israel.
- Psiaki, M. L., Powell, S. P., & Kintner, P. M. (1999, August). The Accuracy of the GPS-derived Acceleration Vector, A Novel Attitude Reference. In *Proceedings of AIAA 1999 Guidance, Navigation, and Control Conference* (p. 751 - 760). Portland, OR.
- Ruijgrok, G. J. J. (1996). *Elements of Airplane Performance* (second ed.). Delft University Press.
- Sarris, Z. (2001, June). Survey of UAV Applications in Civil Markets. In *The 9th IEEE Mediterranean Conference on Control and Automation* (p. 11). Croatia.
- Schmidt, G. T. (2004). INS/GPS Technology Trends. In *Advances in Navigation Sensors and Integration Technology* (p. 1/1 - 16/1).
- Schmidt, G. T., & Phillips, R. (2003, October). INS/GPS Integration Architectures. In *NATO RTO Lecture Series 232* (p. 1 - 15).
- Schultz, C. E. (2006). *INS and GPS Integration*. Unpublished master's thesis, Technical University of Denmark, Lyngby.
- Serrano, L., Kim, D., & Langley, R. B. (2004, 26 - 28 January). A GPS Velocity Sensor: How Accurate Can It Be? A First Look. In *ION NTM 2004* (p. 875- 885). San Diego, California.
- Shin, E. H. (2001). *Accuracy Improvement of Low Cost INS/GPS for Land Applications*. Unpublished master's thesis, University of Calgary.
- Shin, E.-H. (2005). *Estimation Techniques for Low-Cost Inertial Navigation*. Unpublished doctoral dissertation, Dept. of Geomatics Eng., University of Calgary, Calgary, Canada.
- Simon, D. (2006). *Optimal State Estimation: Kalman, H(infinity), and Nonlinear Approaches* (First ed.). Wiley Interscience.
- Skog, I., & Handel, P. (2010). Time synchronization errors in GPS-aided inertial navigation systems. In *IEEE Transactions on Intelligent Transportation Systems*. (In Revision)
- Solimeno, A. (2007). *Low-Cost INS/GPS Data Fusion with Extended Kalman Filter for Airborne Applications*. Unpublished master's thesis, Instituto Superior Tecnico, Lisboa.
- Titterton, D., & Weston, J. (2004). *Strapdown Inertial Navigation Technology* (2nd ed.). Institution of Electrical Engineers.

- Unmanned Dynamics LLC. (2002). Aerosim Blockset - Version 1.2 - Users Guide [Computer software manual].
- Wang, S., Li, Y., & He, W. (2011). Flight attitude estimation for MAV based on M-estimation. In *Consumer Electronics, Communications and Networks (CECNet), 2011 International Conference on* (p. 4968 - 4973).
- Welch, G., & Bishop, G. (1995). *An Introduction to the Kalman Filter* (Tech. Rep.). Chapel Hill, NC: Department of Computer Science, University of North Carolina.
- Wertz, J. R. (1978). *Spacecraft Attitude Determination and Control*. Kluwer Academic.
- Xing, Z., & Gebre-Egziabher, D. (2008, May). Modeling and Bounding Low Cost Inertial Sensor Errors. In *IEEE/ION PLANS 2008* (p. 1122 - 1132). Monterey, CA.
- Xsens Technologies, B. (2009, May). MTi-G User Manual and Technical Documentation [Computer software manual]. Available from [www.xsens.com](http://www.xsens.com) (Document MT0137P.G)
- Xu, F., Zhaooying, Z., Wei, X., & Qi, G. (2008, October). MEMS-Based Low-Cost Flight Control System for Small UAVs. *Tsinghua Science and Technology*, 13(5), 614 - 618.
- Zhang, J. (2005). Real-time Doppler/Doppler Rate Derivation for Dynamic Applications. In *Journal of Global Positioning Systems* (Vol. 5, p. 95-105).

# Appendix A

## Stochastic Processes

Due to the importance of stochastic processes to model sensor error dynamics, a brief explanation of them is provided here.

### A.1 White Noise

The white noise is defined as a stationary random process having a constant power spectral density function, consisting in a sequence of uncorrelated random variables (Shin, 2005). This means that the power is distributed uniformly over all frequency domain. White noise is a very useful approximation in situations in which a disturbing noise is wideband compared with the bandwidth of the system. It is not physically realizable but it is usually used to describe the noise in an inertial sensor.

### A.2 Random Walk

When a white noise process is integrated, a random walk process (also called Wiener processes) is obtained (Grewal et al., 2001). The continuous-time state equation for the random walk is given by:

$$\dot{x}(t) = w(t) \quad (\text{A.1})$$

where  $E[w(t)w(\tau)] = q(t)\delta(t - \tau)$ . In discrete-time we obtain the following:

$$x(k+1) = x(k) + w(k) \quad (\text{A.2})$$

where the noise covariance is  $q(k) = q\Delta t_k$ . The state uncertainty of the random walk increases with time. The Inertial Navigation Systems integrate signals from accelerometers and gyroscopes, where the white noise components are integrated and this will increase the uncertainty of the velocity and attitude parameters, over time.

### A.3 Random Constant

The random constant is a non-dynamic quantity with a fixed, although random, amplitude (Shin, 2005). The continuous and discrete-time processes are described in the next equations, respectively:

$$\dot{x}(t) = 0 \quad (\text{A.3})$$

$$x(k+1) = x(k) \quad (\text{A.4})$$

The random constant is typically used to model the gyroscopes' (or accelerometers') *turn-on* bias, that remains constant during an interval of activity time.

## A.4 Gauss-Markov Process

A stationary Gaussian process that has an exponentially decaying autocorrelation is called a first-order Gauss-Markov process (Shin, 2005; Godha, 2006). This process is characterized by a correlation time  $\tau_c$  and mean squared value  $\sigma^2$  with zero mean. The model is given by:

$$\dot{x}(t) = -\frac{1}{\tau_c}x(t) + w(t) \quad (\text{A.5})$$

for which,  $q = 2\sigma^2/\tau_c$ . The discrete-time model is given by:

$$x(k+1) = e^{-\Delta t_{k+1}/\tau_c}x(k) + w(k) \quad (\text{A.6})$$

with  $q(k) = \sigma^2(1 - e^{-2\Delta t_{k+1}/\tau_c})$ . Thus, in order to establish a Gauss-Markov process the  $\tau_c$  and  $\sigma^2$  parameters have to be known. The Gauss-Markov process is important in engineering sciences for the reason that it seems to fit a large number of physical processes with reasonable accuracy. One of the most important features of this process is that it can represent bounded uncertainty. Therefore, it is used in INS filters to model slowly varying sensor errors. For instance, the bias-drift can be seen as a Gauss-Markov process. However, for low-cost IMUs it is sometimes difficult to obtain the correlation time  $\tau_c$  (Machado, 2011) and the IMUs are only modeled as random walk processes.

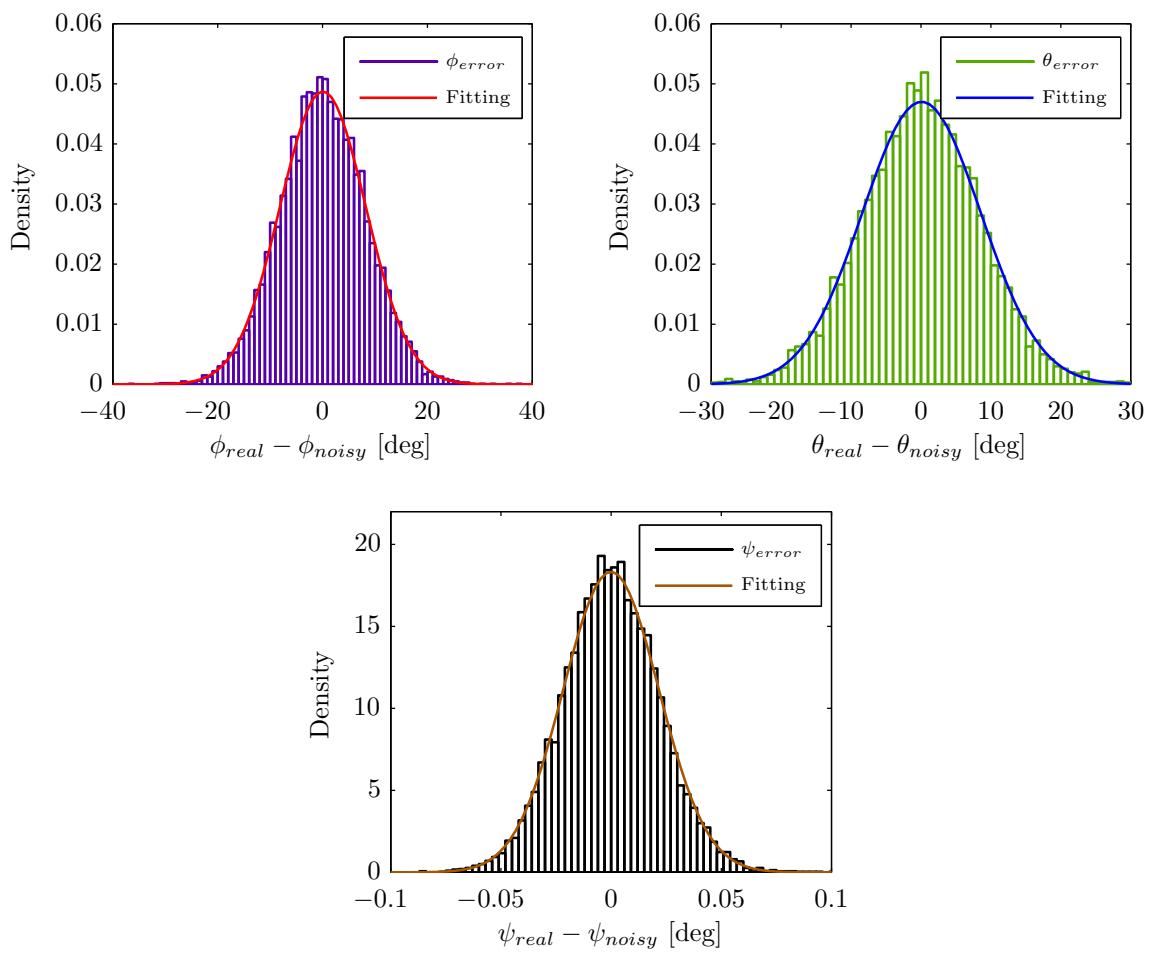
## Appendix B

# Gaussian Distribution Verification

One problem was raised upon the measurements models derivation in Sections 4.2.3 and 4.2.4: the various white noises might no longer keep their *Gaussianity* after undergoing the nonlinear transformations/calculations of Eqs. (4.50) and (4.67). This can be verified in simulation after introducing the noise, described in Section 5.1.3, in the sensors. After introducing the measurements of the sensors in Eqs. (4.50) and (4.67), and with the knowledge of the real attitude angles it is possible to compute error between the true and noisy attitude measurements. Finally, with the help of a Gaussian distribution fitting tool we can verify if the measurements equation can still be given as Eq. (3.29):

$$\underline{z}(t_k) = \underline{z}_m(t_k) + \underline{v}(t_k), \quad k = 1, 2, \dots \quad (\text{B.1})$$

where  $\underline{v}$  is white Gaussian noise. The fitting results for Flight 1 (described in Section 5.1.4) are presented in Figure B.1 for roll, pitch and yaw angles. As we can see, their distribution remains relatively Gaussian after undergoing the nonlinear transformations/calculations of Eqs. (4.50) and (4.67).

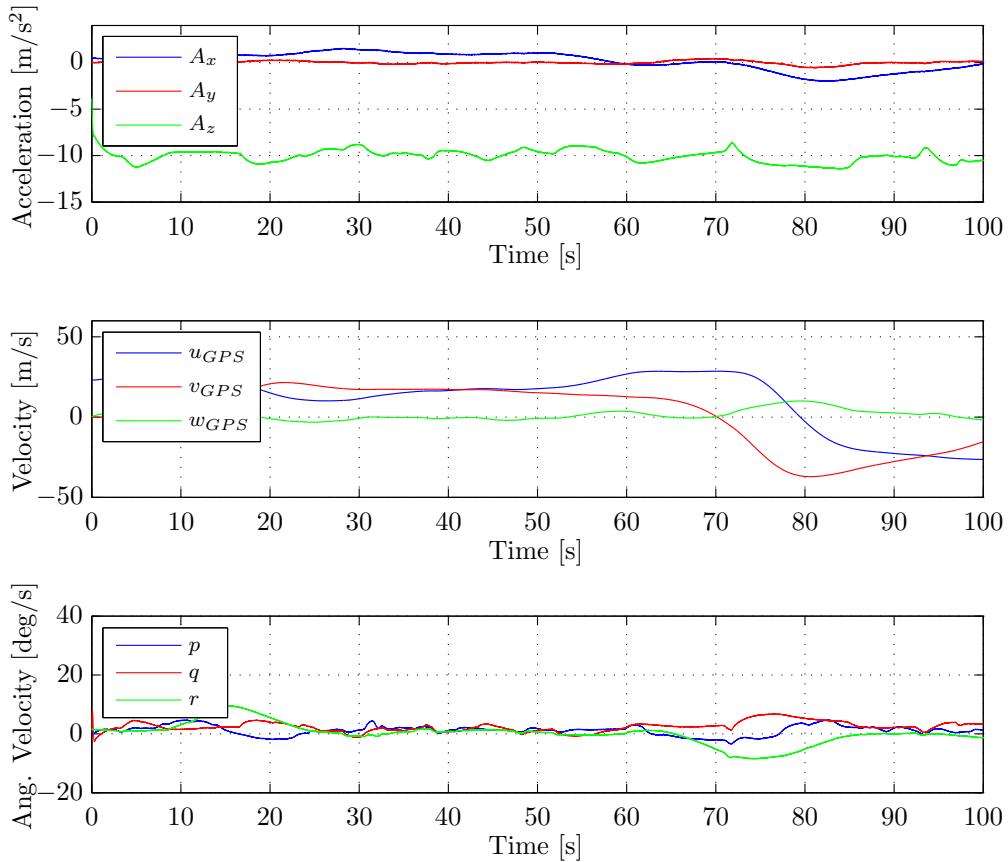


**Figure B.1:** Verification of a Gaussian fitting for the noise of the measurements vector.

## Appendix C

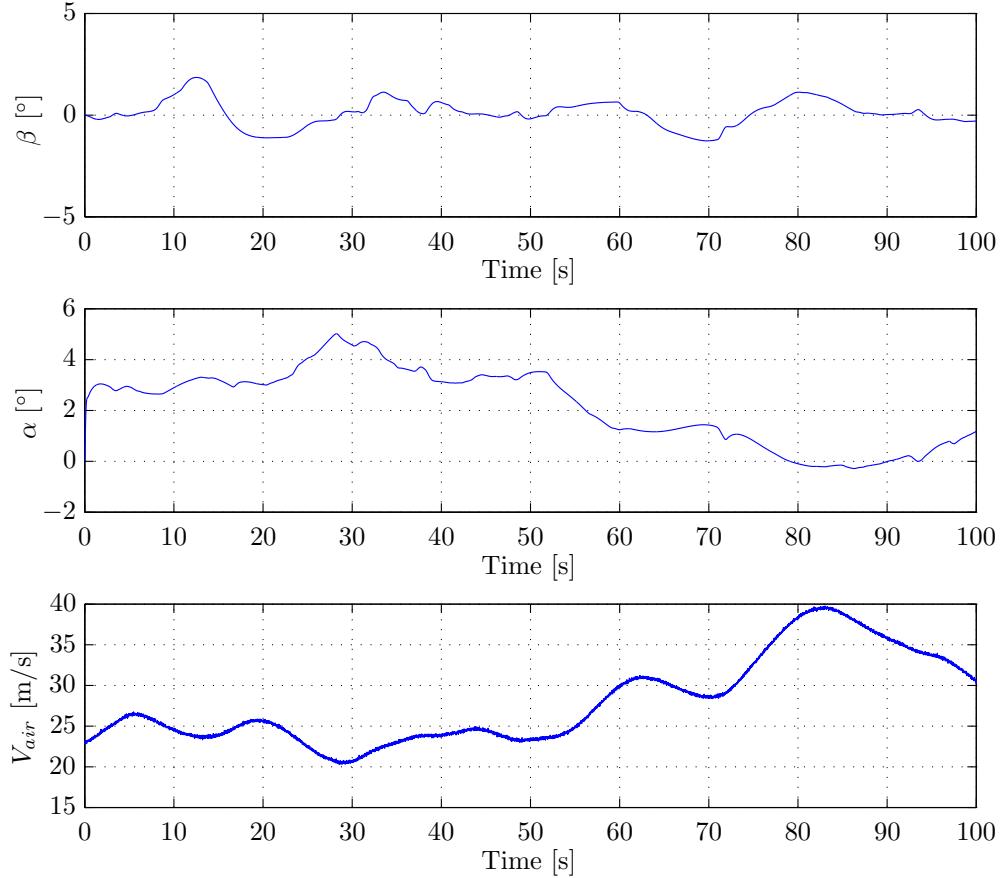
# Simulation Flights Data

In this appendix, the flight data relative to the simulated flights is presented. In Figure C.1, the GPS velocity, accelerometers and gyroscopes measurements of the first flight are presented. The aircraft's trajectory was already described in Figure 5.3.

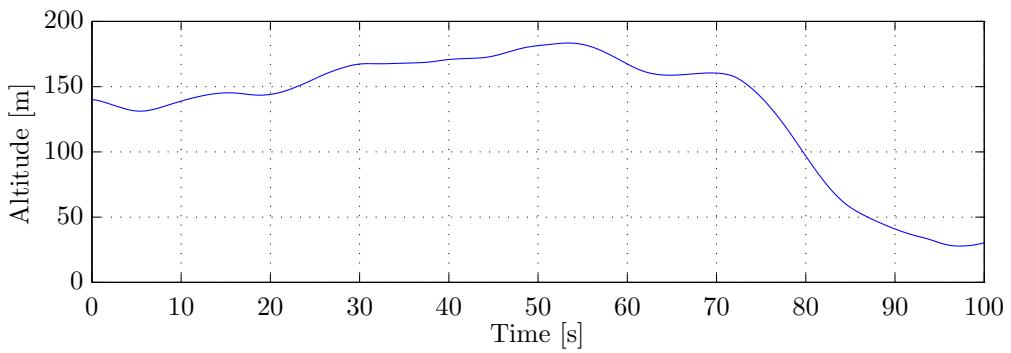


**Figure C.1:** Data from Flight 1: accelerometers, GPS velocity and gyroscopes measurements.

In Figure C.2, the sideslip angle, angle of attack and airspeed are presented. Recall that the sideslip and angle of attack sensors are considered that they do not exist in the UAV, thus we can not use them in the estimation algorithm. Figure C.3 presents the aircraft's altitude.

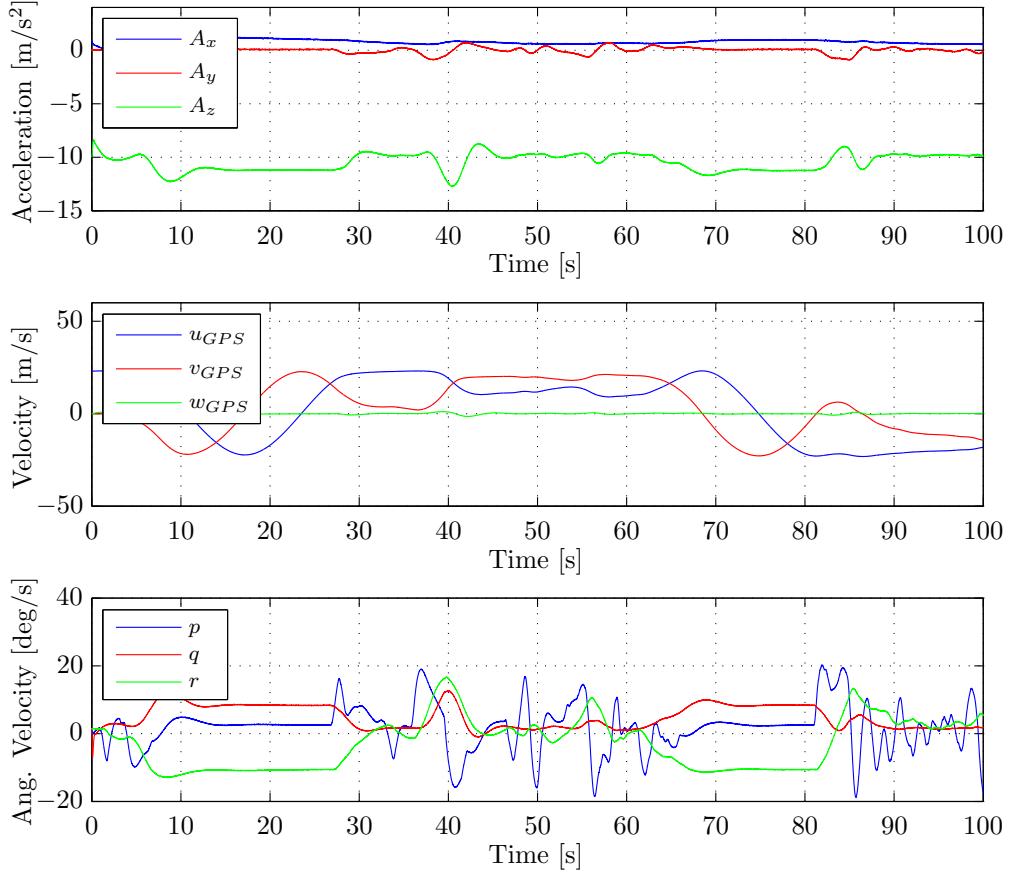


**Figure C.2:** Data from Flight 1: angle of sideslip, angle of attack and airspeed.



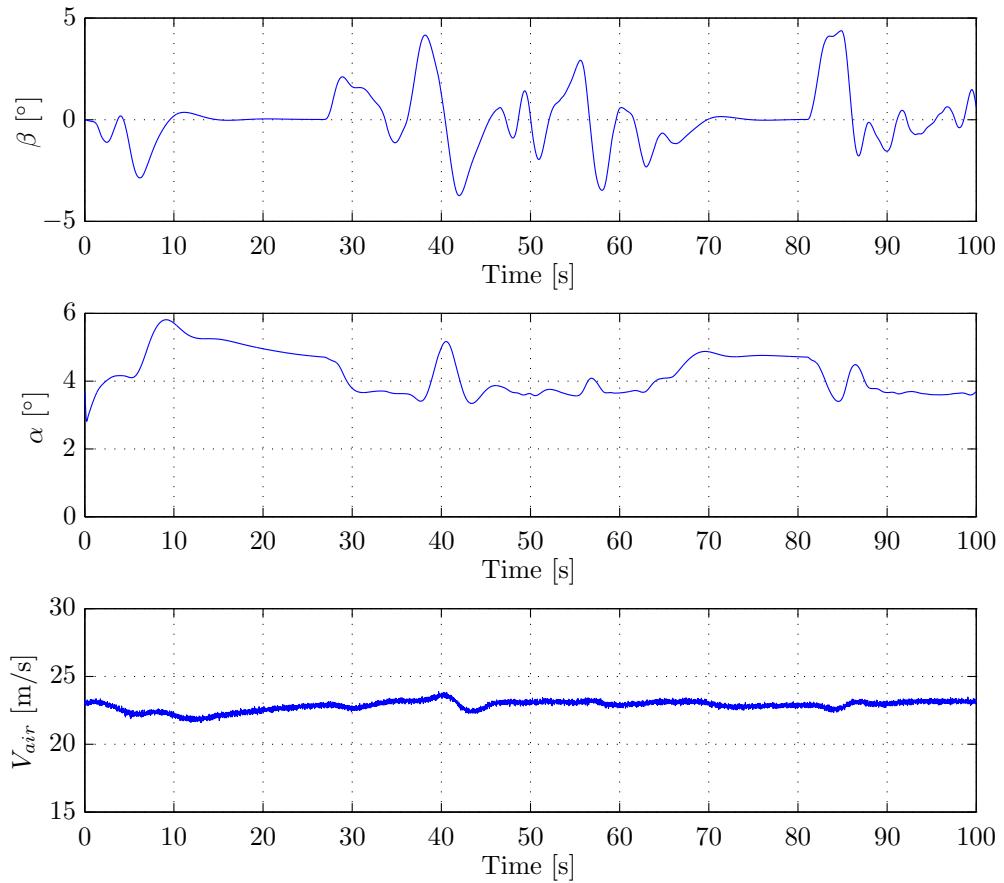
**Figure C.3:** Data from Flight 1: altitude.

The flight data relative to the second simulated flight is presented in the next figures. In Figure C.4, the GPS velocity, accelerometers and gyroscopes measurements are presented. The aircraft's trajectory was already described in Figure 5.4.

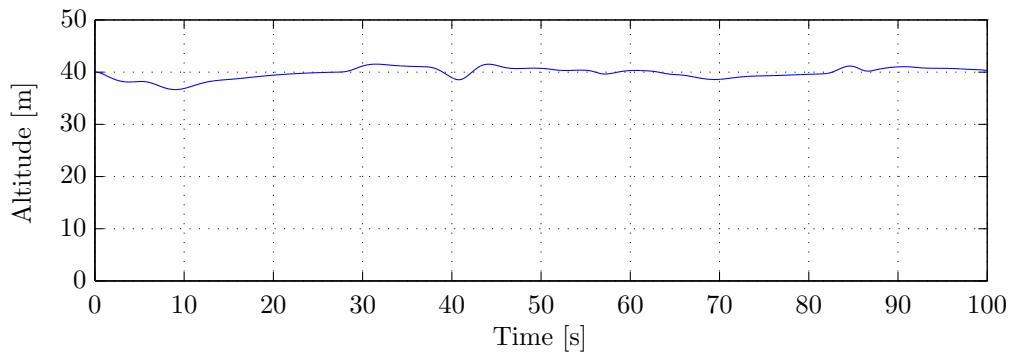


**Figure C.4:** Data from Flight 2: accelerometers, GPS velocity and gyroscopes measurements.

In Figure C.5, the sideslip angle, angle of attack and airspeed are presented. Figure C.6 presents the aircraft's altitude.



**Figure C.5:** Data from Flight 2: angle of sideslip, angle of attack and airspeed.

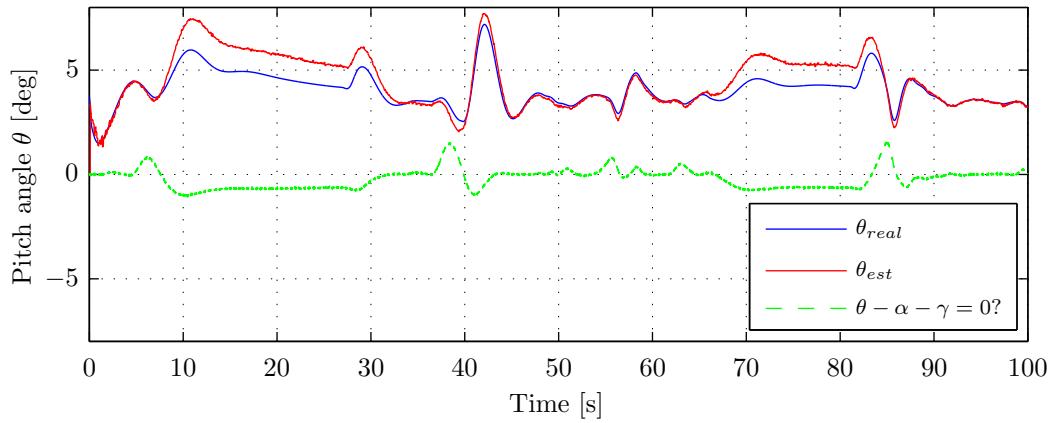


**Figure C.6:** Data from Flight 2: altitude.

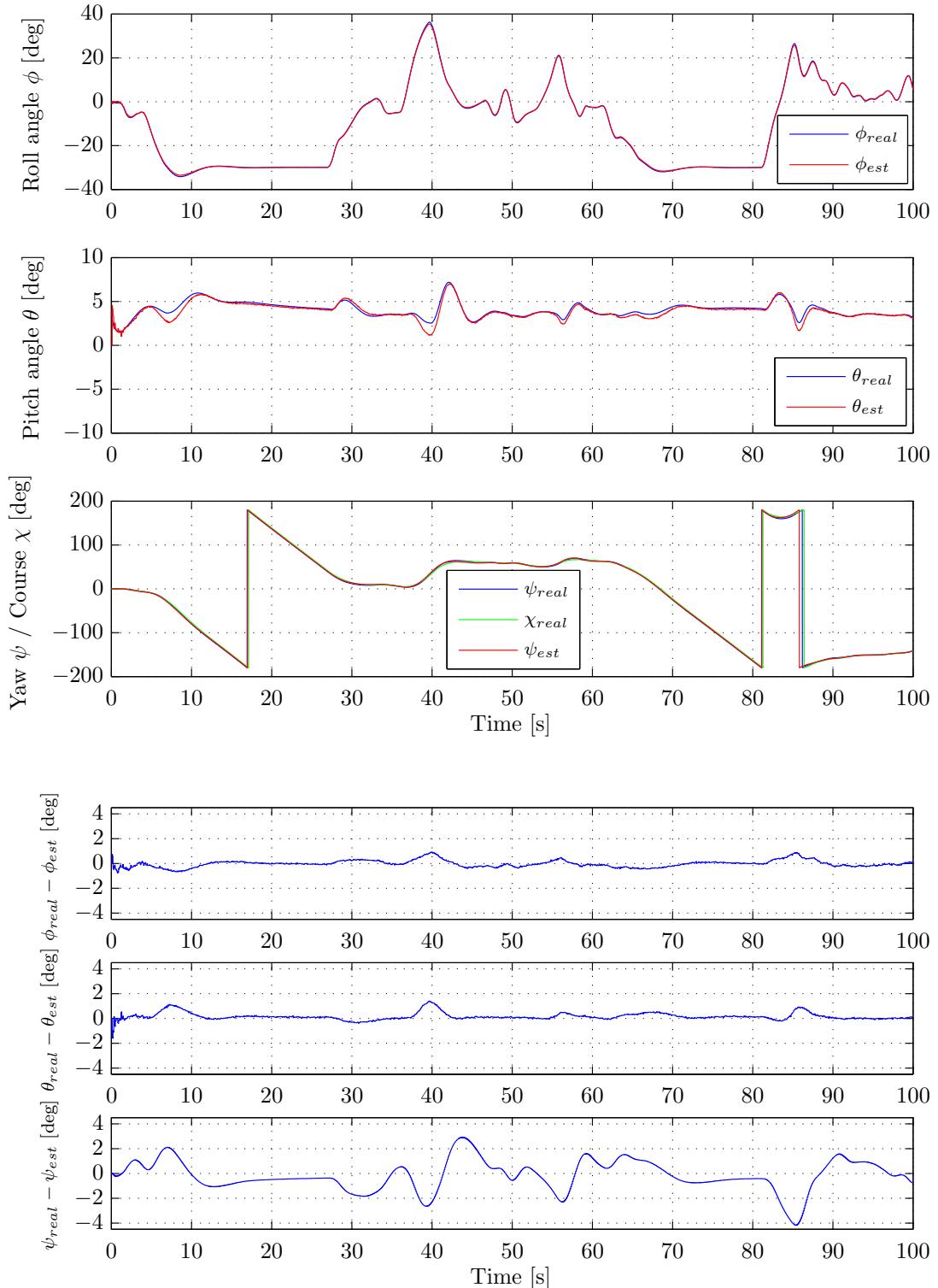
## Appendix D

# Additional Simulation Results

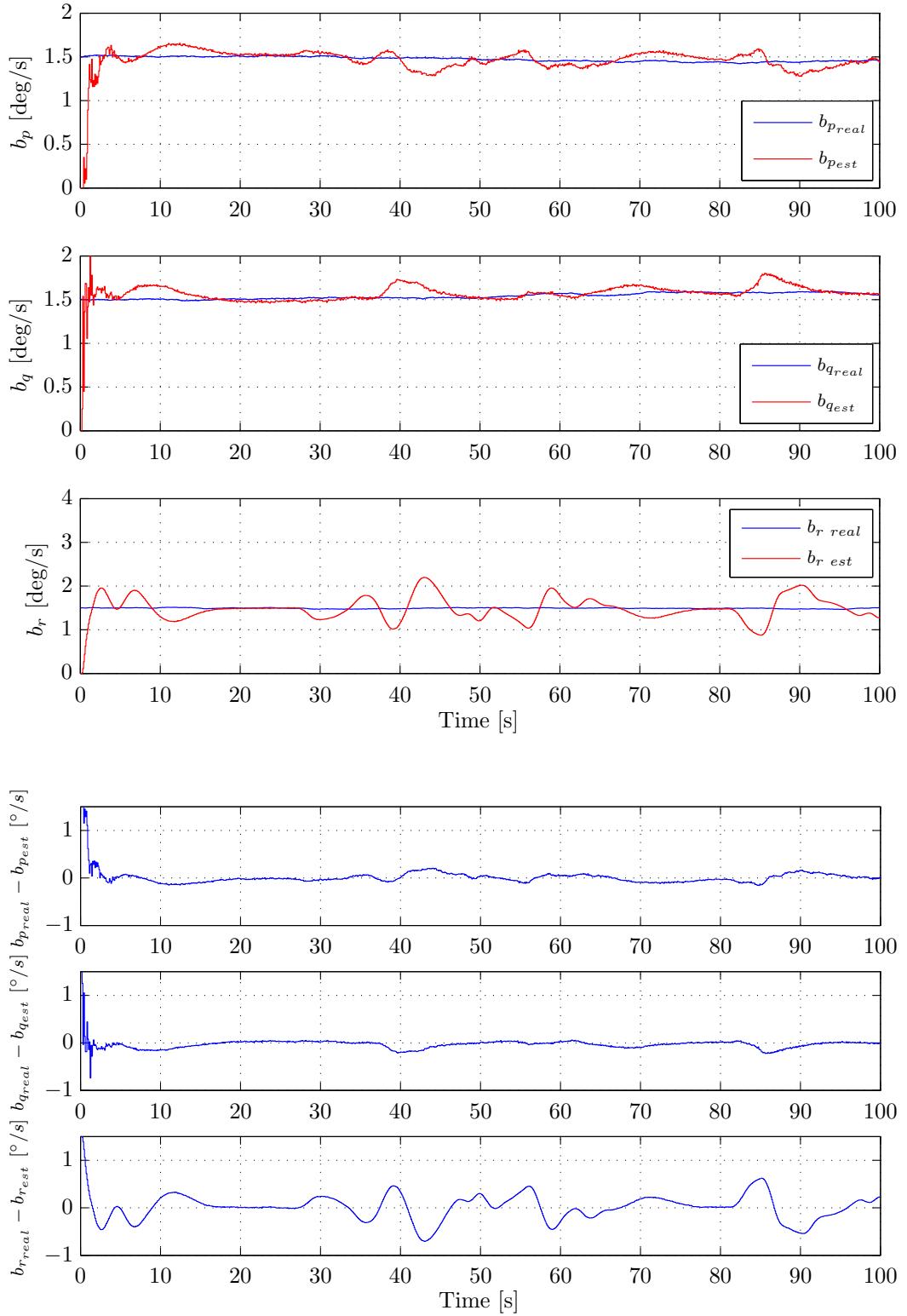
### D.1 EKF<sub>1</sub> and EKF<sub>2</sub>



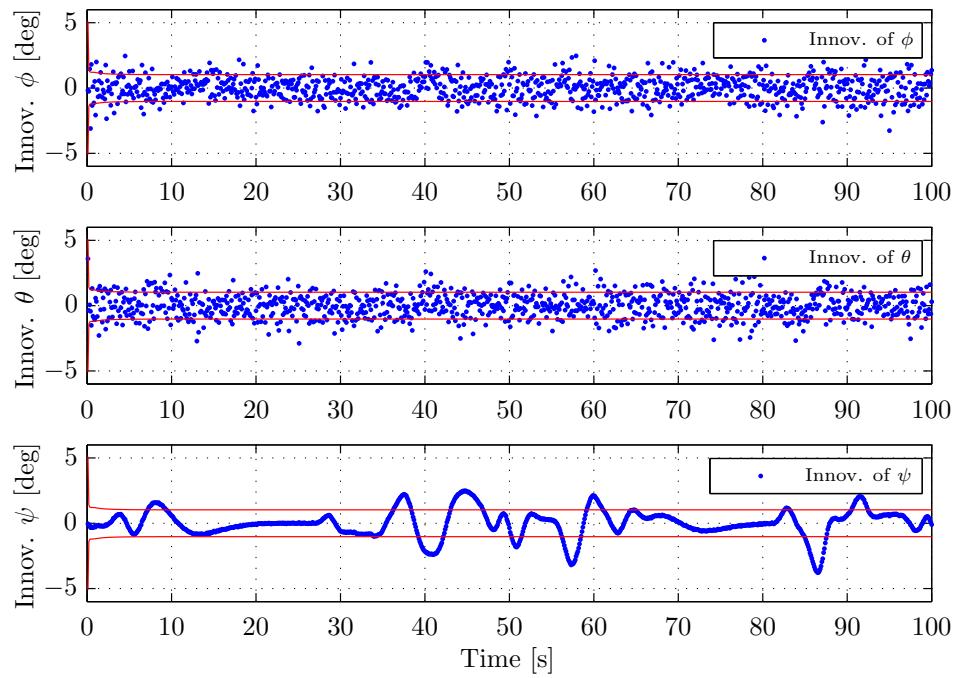
**Figure D.1:** Pitch angle estimation with EKF<sub>1</sub> for Flight 2, with verification of Eq. (2.13).



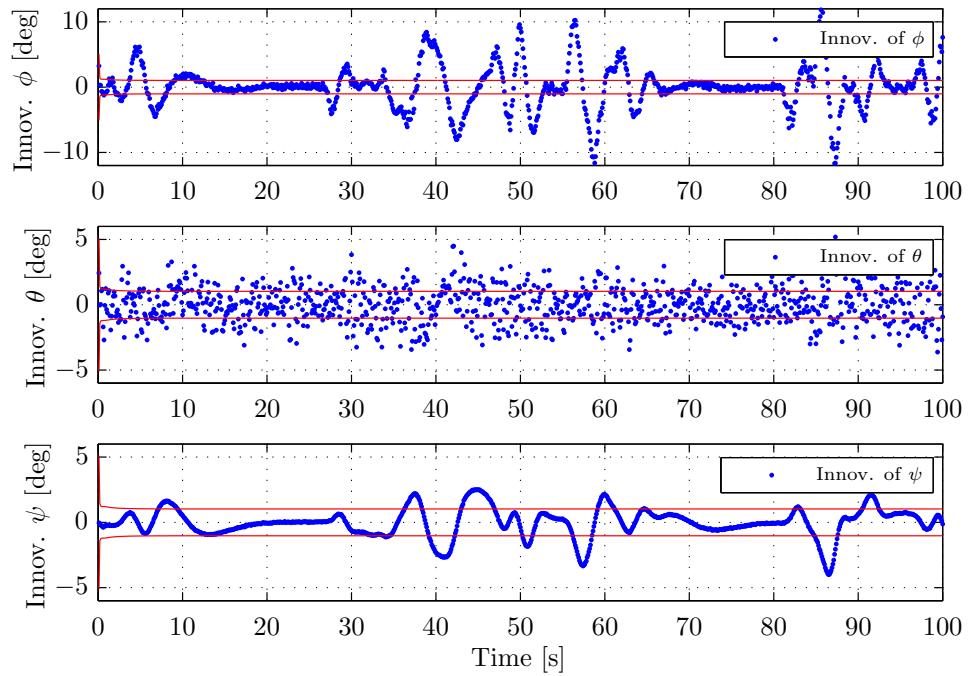
**Figure D.2:** Estimation of  $\phi$ ,  $\theta$  and  $\psi$ , and differences between real and estimated angles, with EKF<sub>1</sub> for Flight 2 with inclusion of Eq. (5.9).



**Figure D.3:** Estimation of  $b_p$ ,  $b_q$  and  $b_r$ , and differences between real and estimated bias with EKF1 for Flight 2 with inclusion of Eq. (5.9).

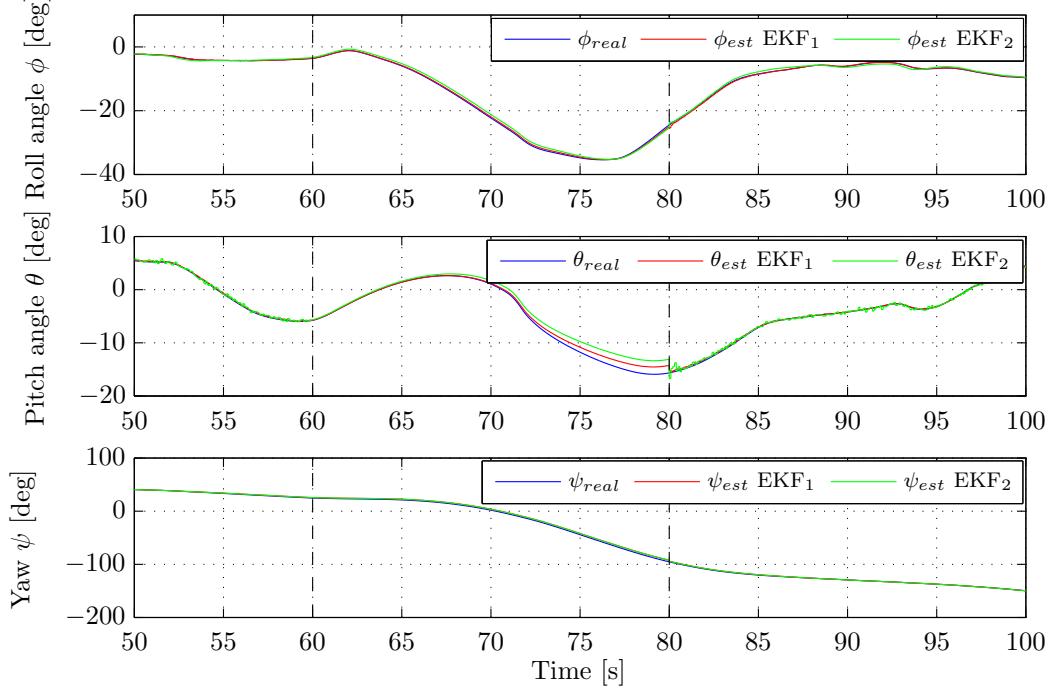


**Figure D.4:** Innovation of the three Euler angles with EKF<sub>1</sub> for Flight 2. The red line represents the standard deviation of the Innovation.

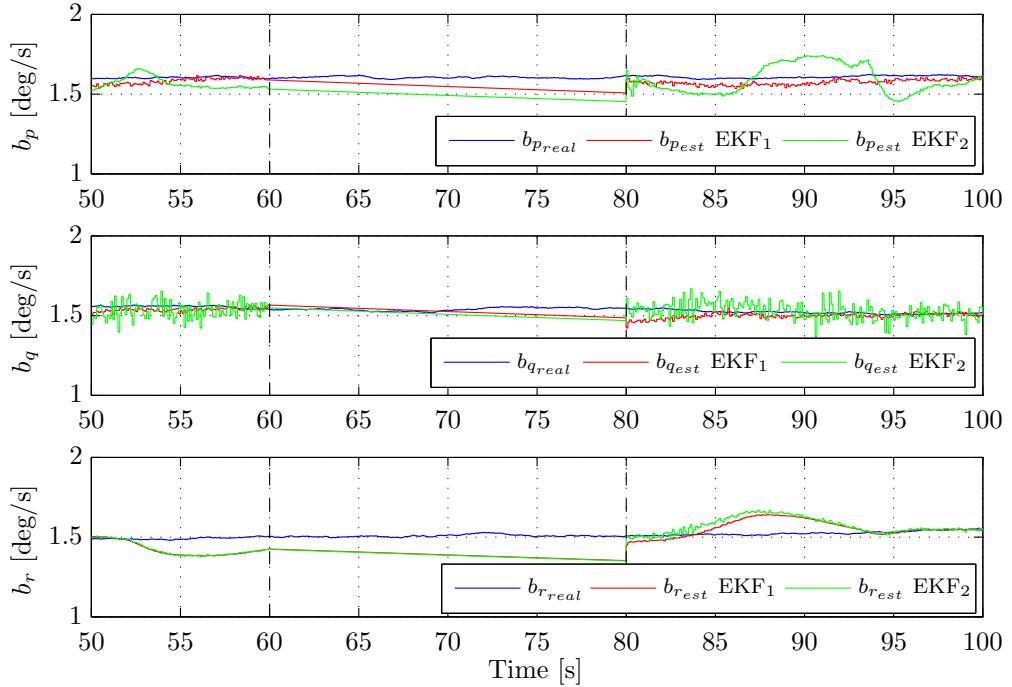


**Figure D.5:** Innovation of the three Euler angles with EKF<sub>2</sub> for Flight 2. The red line represents the standard deviation of the Innovation.

## D.2 GPS Outages



**Figure D.6:** Attitude angles estimation with GPS outages during  $t \in [60, 80]$  (Flight 1).



**Figure D.7:** Estimation of the bias of the gyroscopes with GPS outages during  $t \in [60, 80]$  (Flight 1).

# Appendix E

## Real Data

### E.1 MTi-G - Xsens Device

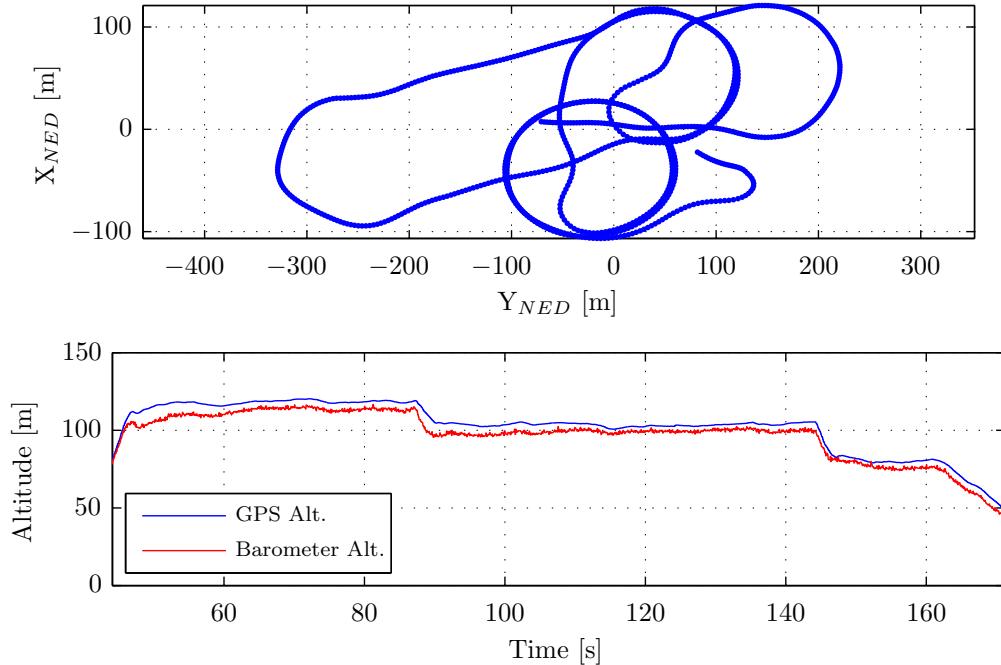
The MTi-G is an integrated GPS and MEMS Inertial Measurement Unit with a Navigation/AHRS processor. The internal low-power signal processor runs a real-time Xsens Kalman Filter providing inertial enhanced 3D position and velocity estimates. The MTi-G also provides drift-free, GPS enhanced, 3D orientation estimates, as well as calibrated 3D acceleration, 3D rate of turn, 3D earth-magnetic field data and static pressure (barometer). The most important characteristics of the MTi-G's accelerometers and gyroscopes are presented in Table E.1. For more information about the device refer to (Xsens Technologies, 2009). With the default configuration, the MTi-G outputs data at a frequency of 100Hz. The GPS update rate is of 4Hz.

The Xsens Kalman Filter filter inside the MTi-G is a causal filter. After turning on, the MTi-G needs some time, typically 30 seconds or more, to stabilize. Still, the filter will continue to improve even after 15 minutes. This time is required to establish a good position, orientation and rate gyroscope bias. Additionally, the Xsens Kalman Filter navigation algorithm inside the MTi-G is *loosely coupled* meaning that the position and velocity estimates from the GPS receiver unit, contained in the MTi-G device, are then used in a *global* Kalman Filter.

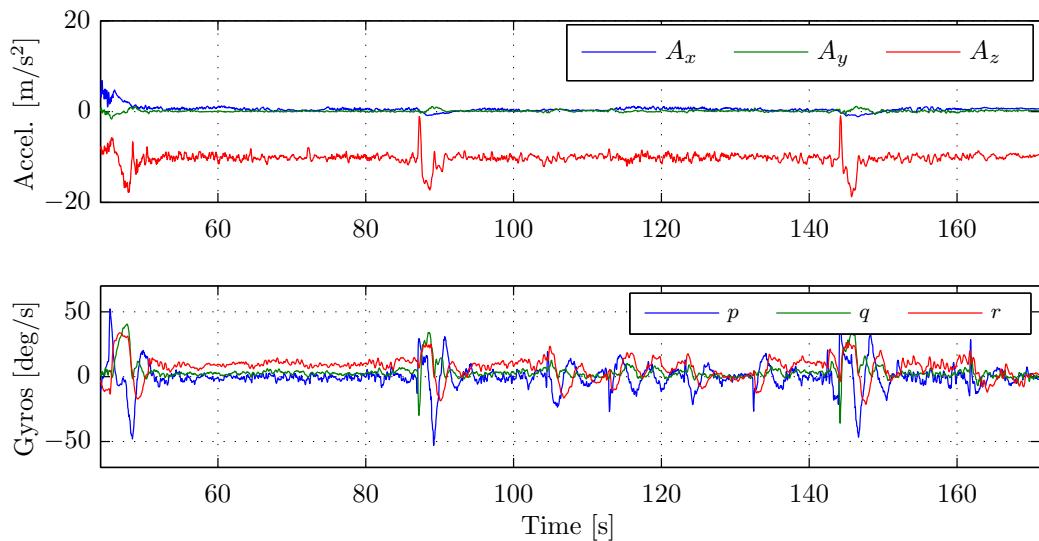
**Table E.1:** Calibrated inertial data performance for a MTi-G device with standard configuration.

	Gyroscopes	Accelerometers
Units	[deg/s]	[m/s <sup>2</sup> ]
Bias Stability [1 $\sigma$ ]	1	0.02
Scale factor stability [% 1 $\sigma$ ]	-	0.03
Noise density [units/ $\sqrt{\text{Hz}}$ ]	0.05	0.002
Alignment error [deg]	0.1	0.1
Bandwidth [Hz]	40	30

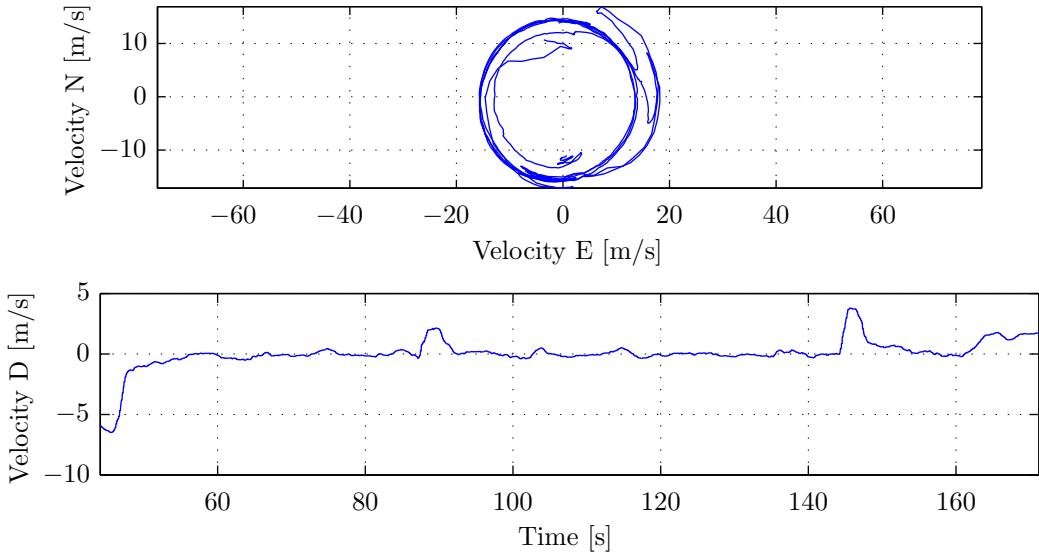
## E.2 Additional Data of Real Flight



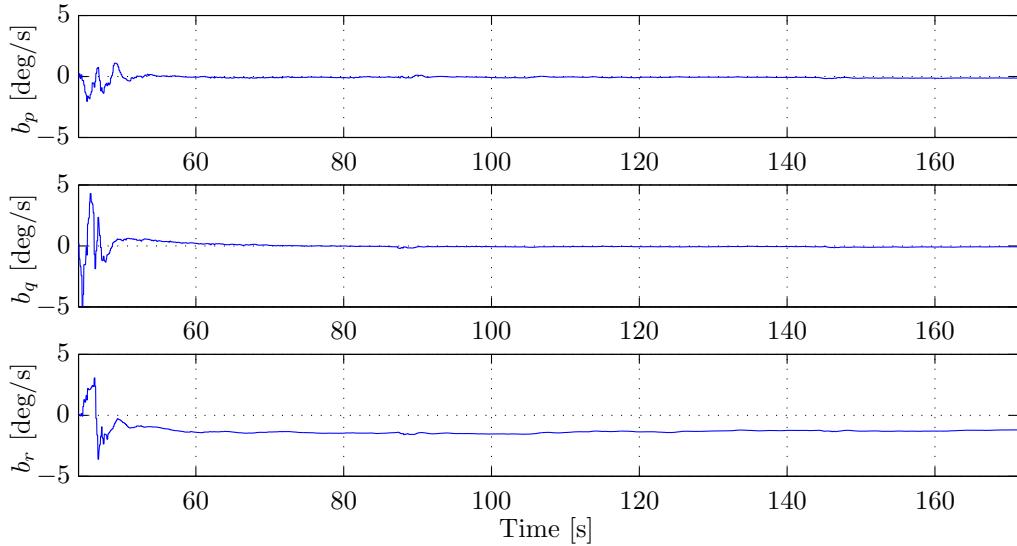
**Figure E.1:** Real Flight Data (LOG00060) - North-East position (GPS) and Altitude (GPS-Barometer).



**Figure E.2:** Real Flight Data (LOG00060) - accelerometers and gyroscopes measurements.



**Figure E.3:** Real Flight Data (LOG00060) - GPS velocity NED.



**Figure E.4:** Estimation of the gyroscopes' bias for the Real Flight.

### E.3 Tunning Parameters

The initial Error covariance matrix  $P(0|0)$  was chosen with higher values (than in the simulation) for the attitude angles:

$$P(0|0) = \text{diag} \left( \left[ \left( \frac{10\pi}{180} \right)^2, \left( \frac{10\pi}{180} \right)^2, \left( \frac{90\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2 \right] \right) \quad (\text{E.1})$$

The initial states estimation  $\hat{x}(0|0)$  can be defined null, as in the simulation chapter:

$$\hat{x}(0|0) = [0, 0, 0, 0, 0, 0]^T \quad (\text{E.2})$$

The parameter  $\lambda$  introduced in Section 5.1.4 was reset to its original value:

$$\lambda = 1 \quad (\text{E.3})$$

In a real flight the  $Q$  and  $R$  matrices have to be tuned manually in order to achieve proper results. Nevertheless, the simulation values can give an acceptable initial search point. The process noise covariance matrix  $Q$  used was:

$$Q = \text{diag} \left( \left[ \left( \frac{0.05\pi}{180} \right), \left( \frac{0.05\pi}{180} \right), \left( \frac{0.05\pi}{180} \right), 1 \times 10^{-8}, 1 \times 10^{-8}, 1 \times 10^{-8} \right] \right) \quad (\text{E.4})$$

On the other hand, the measurements noise covariance matrix  $R$  is given by:

$$R = \text{diag} \left( \left[ \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2, \left( \frac{2\pi}{180} \right)^2 \right] \right) \quad (\text{E.5})$$